MICROHOBBY SEMANAL

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

AÑO II- N.º 13

95 PTAS. Canarias 105 ptas.

EDITA HOBBY PRESS, S.A.



SOFTWARE

JET SET WILLY, DESVENTURAS DE UN MINERO RICO

EDITOR DE TEXTOS PROFESIONAL PARA EL SPECTRUM

¡CON 64 CARACTERES POR LINEA!

PROGRAMAS

- CAMPOS DE ENERGIA
- HELP
- EL BOMBARDERO

HARDWARE

INTERIORIDADES Y FUNCIONAMIENTO DE LA "ULA"

Franqueo
Postal

HOBBY PRESS, S. A.

Apartado de Correos

n.º **54.062** (Apartados Altos)

MADRID

Director Editorial
José I. Gómez-Centurión
Director Ejecutivo
Domingo Gómez
Redactor Jefe
Africa Pérez Tolosa
Diseño
Jesús Iniesta
Maqueta
Rosa M. Capitel
Redacción
José María Díaz
Gabriel Nieto
Colaboradores
Jesús Alonso, Lorenzo Cebeira, Primitivo de Francisco, Rafael Prades, Víctor Prieto
Fotografía
Javier Martínez
Carlos Candel
Portada
José María Ponce
Dibujos
Manuel Berrocal, J.R. Ballesteros, A. Perera, F.L. Frontán, J. Septien, J.M. López Moreno
Edita
HOBBY PRESS, S.A.
Presidente
María Andrino
Consejero Delegado
José I. Gómez-Centurión
Administrador General
Ernesto Marco
Jefe de Publicidad
Marisa Esteban
Secretaria de Publicidad
Concha Gutiérrez
Publicidad Barcelona
Isidro Iglesias
Tel.: (93) 307 11 13
Secretaria de Dirección
Marisa Cogorro
Suscripciones
M.ª Rosa González
M.ª del Mar Calzada
Redacción, Administración y Publicidad
La Granja, n.º 8
Polígono Industrial de Alcobendas
Telf.: 654 32 11
Distribución
Coedis, S.A. Valencia, 245.
Barcelona.
Imprime
Rotedic, S.A.
Carretera de Irún, Km. 12,450
Tel.: 734 15 00
Fotocomposición
Consulgraf
Nicolás Morales, 34 - 1.º
Tel.: 471 29 08
Fotomecánica
Zescán
Nicolás Morales, 38
Tel.: 472 38 58
Depósito Legal:
M-36.598-1984
Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América, 1.532. Telf.: 21 24 64. 1209 BUENOS AIRES (Argentina).
Derechos Exclusivos
«Sinclair Users», «Sinclair Programs» y «Sinclair Projects» de EMAP Publications (Londres).

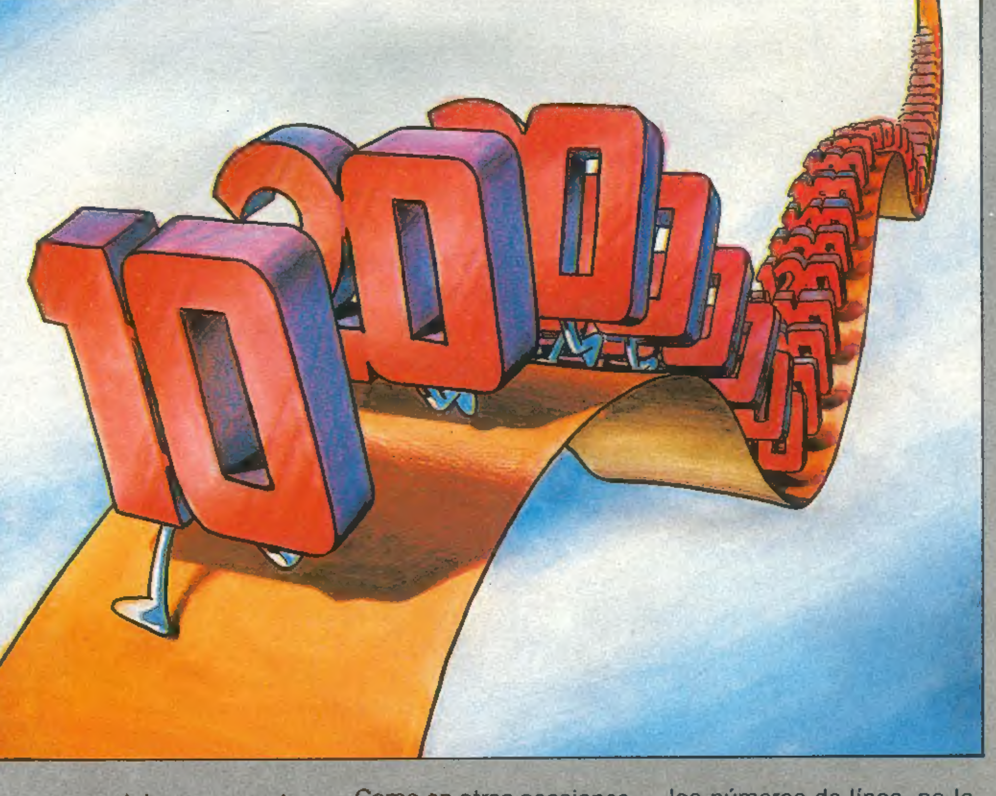
Se solicitará control OJD

MICROHOBBY ESTA SEMANA

Año II - N.º 13 - 29 de enero al 4 febrero de 1985
95 ptas. (Sobretasa Canarias 10 ptas.)

TRUCOS

RENUMERADOR DE LINEAS DE PROGRAMA



Cuando finalizamos un programa Basic, normalmente aparece como un auténtico caos en cuanto a la numeración de las líneas.

Por ello hemos construido un modesto renumerador, en lenguaje máquina, que nos permitirá colocarlas por orden a partir de un determinado número de líneas y con un «paso» entre ellas a elegir.

El ordenador utiliza los primeros cuatro bytes de cada línea para almacenar en ellos su número de orden (los dos primeros) y la longitud del texto más «ENTER»; es decir, si la longitud fuera de diez bytes, se almacenaría el valor 11.

Basta, por tanto, poner en las dos primeras posiciones de memoria de cada línea de programa, el nuevo valor inicial más el paso, número que será acumulativo para la siguiente; así nos vamos recorriendo toda la longitud del programa hasta alcanzar el área de variables. Se detecta esta circunstancia finalizando el cálculo y saltando a la rutina ROM que provoca el listado del programa Basic ya renumerado.

Hemos incluido un pequeño programa cargador en Basic con la doble idea de aclarar cómo se introducen, en posiciones sucesivas de memoria, un número mayor que 255, y de que el lector experimente renumerando el propio cargador.

Si no se desea emplear el programita y sólo se trata de renumerar de diez en diez, por ejemplo, puede hacerse mediante POKE de la siguiente manera:

POKE 23337,Ø
POKE 23339,1Ø

(seguido de RANDOMIZE USR 233ØØ) o cualquier otro par de números menores de 255.

Como en otras ocasiones, la rutina en lenguaje máquina se ha colocado en la memoria intermedia de la impresora.

Esta rutina sólo renumera los números de línea, no lo hace con los comandos GOTO y GOSUB; además, presupone la existencia de, al menos, una línea de programa Basic.

```
   1 DATA 205,175,13,237,91,83,9
2,42,41,91,237,75,43,91,9,34,43,
91,235,114,35,115,35,78,35,70,9,
35,62,64,190,218,249,23,235,24,2
26,10,0,0,0,255

   9 FOR i=23300 TO 23341: READ
x: POKE i,x: NEXT i
  10 LET ofset=23337: LET nline=
23339
  20 INPUT "Primera linea: ";pli
n,"Interlineas: ";ilin
  30 LET a=INT (ilin/256): LET b
=ilin-(a*256)
  40 LET c=INT (plin/256): LET d
=plin-(c*256)
  50 POKE ofset,b: POKE ofset+1,
a: POKE nline,d: POKE nline+1,c
  60 RANDOMIZE USR 23300
```

PARPADEO, COLOR Y SONIDO

De parte de uno de nuestros lectores nos llega un simpático truco que imita a uno de los efectos que hemos visto en algunos programas comerciales, hecho en Basic y simple como todas las buenas ideas. Cuando el programa pregunta por la repetición del efecto, hay que contestarle con un número.

Truco enviado por el lector Angel Luis MORO SIERRA.

```
  10 BORDER 2: PAPER 2: CLS
  20 INPUT AT 11,4;AT 0,4;"REPET
ICION DEL EFECTO?";a
  30 FOR b=1 TO a
  40 FOR c=0 TO 7
  50 PRINT AT 11,4; INK c;"PARPA
DEO,COLOR Y SONIDO"
  60 BEEP .01,RND*b+12
  70 BEEP .01,RND*b-12
  80 PAUSE 6
  90 NEXT c
 100 NEXT b
 110 PRINT AT 11,0; INK 7;"PARA
LOS AMIGOS DE MICROHOBBY","DE AN
GEL LUIS MORO SIERRA"
```

En este espacio también tienen cabida los trucos que nuestros lectores quieran proponer.
Para ello, no tienen más que enviarlos por correo a MICROHOBBY La Granja, 8. Polígono Industrial de Alcobendas. MADRID.

Para el tratamiento de textos en el Spectrum

EDITEXT (I)

Jesús ALONSO RODRIGUEZ

Una de las aplicaciones más interesantes de los ordenadores domésticos es el tratamiento de textos; y si hay un ordenador especialmente poco idóneo para esta aplicación, es sin duda el Spectrum; el teclado es lento, trabaja con sólo 32 columnas, y por si fuera poco, el BASIC interpretado es un lenguaje demasiado lento para las tareas que ha de llevar a cabo un editor de textos.

En primer lugar, hay que conseguir aumentar el número de columnas en pantalla, para lo cual se genera un nuevo juego de caracteres que permite 64 columnas de impresión; a continuación se diseñan algunas rutinas en código máquina para desarrollar tareas en las que el BASIC resulta lento, tales como transferencia de bloques, impresión de pantallas, etc.; y finalmente se escribe el resto del programa en BASIC. El resultado es el Editor de textos que presentamos aquí con el título de EDITEXT.

Composición del programa

El programa tiene dos partes fundamentales; la primera es un editor de página en curso o página cero, que incorpora una serie de facilidades tales como tabulación, párrafo adentrado, borrado de caracteres, ajuste automático de línea, caracteres españoles, acentuado de vocales, etc.

La segunda parte del programa la componen una serie de rutinas que permiten almacenar en memoria hasta 21 páginas de texto, leer las páginas almacenadas, pasarlas a impresora, o almacenarlas en cintas de cassette. Para todo esto se utilizan cinco rutinas en código máquina; la primera no es preciso comentarla ya que es ampliamente conocida por todos los usuarios del Spectrum; de las restantes se acompaña el listado assembler con un gran número de comentarios que permitirán al lector entender su funcionamiento.

El manejo del Editor no es complicado, después de la «carátula» aparece una página de instrucciones que recuerda los distintos comandos del editor de página en curso, aunque es más sencillo colocar sobre el teclado una plantilla como la que se adjunta. Las instrucciones están escritas sobre la página cero, por lo que se borran al utilizar el editor; no obstante, si desea guardarlas, siempre puede transferir la página en curso a la página 21 utilizando la opción 2 (observe la velocidad con que realiza la transferencia, 2Ø milisegundos); por lo demás, el programa tiene la buena costumbre de indicar lo que va haciendo y pedir los datos que necesite.

La zona de ajuste corresponde a los últimos caracteres de la línea, se puede fijar entre uno y ocho, o cero si no se desea ajuste automático, el programa emite una nota aguda para indicar que ha entrado en zona de ajuste, y que por tanto, la pulsación de un guión (signo «menos») o un espacio, producirá un salto automático a nueva línea; CAPS SHIFT + 2 fija y libera las mayúsculas; CAPS SHIFT + Ø borra el último carácter de la línea en curso; SIMBOL SHIFT + 2 genera la ñ minúscula; SIMBOL SHIFT + 6 genera la Ñ mayúscula; ENTER salta a nueva línea; y CAPS SHIFT + SIMBOL SHIFT pasan a modo extendido.

Al situarse el teclado en modo extendido aparece la palabra EXT. en el ángulo inferior derecho de la pantalla; en este modo puede usarse el tabulador, el párrafo adentrado, el final de página, y las vocales acentuadas.

```
COMANDOS DEL EDITEXT

COMANDOS EN MODO NORMAL      *    COMANDOS EN MODO EXT.
                             *
CAPS SHIFT + 2 = CAPS LOCK   * 1 = FIJAR TABULADORES
                             *
CAPS SHIFT + 0 = BORRADO     * 9 = BORRAR TABULADORES
                             *
ENTER = NUEVA LINEA          * T = TABULADOR
                             *
SIMBOL SHIFT + 2 = ñ         * P = FIJA PARRAFO ADENTRADO
                             *
SIMBOL SHIFT + 6 = Ñ         * 8 = BORRA PARRAFO ADENTRADO
                             *
CAPS + SIMBOL = MODO EXT.    * ENTER = FIN DE PAGINA
                             *
                             * VOCAL = VOCAL ACENTUADA
                             *

      * * PULSE CUALQUIER TECLA PARA EMPEZAR * *
```

Manejo del tabulador

Para usar el tabulador habrá primero que fijar los topes de tabulación; pase a modo extendido y pulse la tecla «1»; se encuentra ahora en rutina de tabulado, ahora lleve el cursor hasta el primer punto que desee tabular utilizando la tecla espaciadora; para fijar la tabulación pulse «T»; repita hasta un máximo de cuatro topes de tabulación; cuando haya fijado el cuarto se producirá un retorno automático al inicio de línea y saldrá de la rutina de tabulado; si desea fijar menos de cuatro topes pulse ENTER para salir de la rutina de tabulado y retornar al modo normal; es posible que su trabajo habitual le obligue a utilizar más de cuatro topes de tabulado. Esperamos que no encuentre problemas en alterar las líneas 7Ø1Ø, 7Ø7Ø, 715Ø, 741Ø, 743Ø y 8Ø2Ø; el array «t(4)» almacena los topes de tabulación entre Ø y 63, y la variable «ct» los subíndices del array «t(4)»; una vez fijados los topes de tabulación cada vez que pulse «T» en mo-

PROGRAMA 1

```
   1 REM ********************
   2 REM *                  *
   3 REM *  CARGADOR  DE    *
   4 REM *                  *
   5 REM * CODIGO MAQUINA   *
   6 REM *                  *
   7 REM *  PARA EDITEXT    *
   8 REM *                  *
   9 REM ********************
  10 CLEAR 32255
  20 RESTORE 1000: LET bl=1
  30 READ d,l
  40 IF d=0 THEN GO TO 500
  50 LET a$=""
  55 FOR f=1 TO 8
  60 READ b$
  65 LET a$=a$+b$
  70 NEXT f
  75 READ s
  80 GO SUB 200+100*(l=768)
  90 LET cs=0
 100 FOR n=d TO d+l-1
 110 LET cs=cs+PEEK n
 120 NEXT n
 130 IF cs<>s THEN GO TO 400
 140 LET bl=bl+1
 150 GO TO 30
 200 FOR n=1 TO LEN a$-1 STEP 2
 210 LET a=CODE a$(n): LET b=COD
E a$(n+1): LET aa=a-48*(a>47)-7*
(a>64): LET bb=b-48*(b>47)-7*(b>
64): LET a=aa*16+bb
 220 POKE d+(n-1)/2,a
 230 NEXT n
 240 RETURN
 300 FOR n=1 TO LEN a$
 310 LET a=CODE a$(n): LET aa=a-
48*(a>47)-7*(a>64)
 320 POKE d+n-1,aa: POKE d+n-769
,aa*16
 330 NEXT n
 340 RETURN
 400 PRINT "ERROR EN BLOQUE ";bl
;" REVISE DATAS"
 410 BEEP .5,0: PAUSE 100
 420 LIST bl*1000
 430 STOP
 500 SAVE "cm"CODE 32256,1986
 510 SAVE "cmp"CODE 65218,292
 520 PRINT "CINTA EN 'PLAY' PARA
 VERIFICAR"
 530 VERIFY "cm"CODE 32256,1986
 540 VERIFY "cmp"CODE 65218,292
 550 CLS : PRINT "CORRECTO"'"HAG
A 'NEW' Y TECLEE EL BASIC": STOP
1000 REM BLOQUE 1
1005 DATA 32256,277
1010 DATA "210F5B7E2322005B6F3CC
8260029292 9ED4B365C093E083204 5B3
A085B32095B3A0A5B"
1020 DATA "32085B3E0932055B7E232
2025B0732065B3A055B3D20323A045B3
D20183A0E5B473A0C"
1030 DATA "5B4F3A0A5B810520FC320
A5B2A005BC3037E32045B3A0D5B473A0
95B8032095B2A025B"
1040 DATA "C3207E32055B3A0C5B473
A095B32075B3A0D5B4FC5CDA47EC13A0
75B3C32075B0D20F1"
1050 DATA "3A085B3C32085B0520DD3
A065BC3307E8040201008040201 3A8E5
CEEFF473A8D5CA047"
1060 DATA "3A085BE6F86F3A075BFEC
0D01F1F1FE61F67CB1CCB1DCB1CCB1DC
B1CCB1D3E58B4673A"
1070 DATA "8E5CA6B0773A075B47E60
7F64067781F1F1FE618B467781717E6E
06F3A085B471F1F1F"
1080 DATA "E61FB56FEB219C7E78E60
74F06000946 1A21065BCB462803B012C
92FB02F12C9"
1090 DATA 21932
2000 REM BLOQUE 2
2005 DATA 32589,117
2010 DATA "3E0DD72318083E02CD011
621C2853E"
2020 DATA "C232365C3E7E32375C7EF
EFF283C7E"
2030 DATA "FE0D28DE7EF600282AD73
E08D7233E"
2040 DATA "8132375C7EFEFF28237EF
E0D28C57E"
2050 DATA "F6002815D723E521428BC
1373FED42"
2060 DATA "280CC5E118BD3E2018D23
E2018E7C9"
2070 DATA "1142F921C285018005EDB
0C92142F9"
2080 DATA "11C285018005EDB0CD537
FC9"
2090 DATA 12577
3000 REM BLOQUE 3
3005 DATA 33474,768
3010 DATA "0000000004444040 0AA00
000004E4E404E8C62E400924900609DB
990240000002488884284222248000A4
A000004E400"
3020 DATA "00002240000 0E00000000
02000124800 0EAAAE0026A22200E22E
8E00E22E2E00AAE22200E8E22E00C88E
AE00E222220"
3030 DATA "0EAAEAE00EAAE22000400
40002002240012484210 00E0E0008421
2480EA24040060CAAA00EAAEAA00CACA
AC00EA88AE0"
3040 DATA "0CAAAAC00E8C88E00E8C8
8800E88AAE00AAEEAA004444440 02222
AE00ACCAAA008888 8E00AEEAAA009DB9
99004AAAA40"
3050 DATA "0EAE88004AAAE610CAAC
A900EA842E00E444440 0AAAAAE00AAAA
A400AAAAEE00AA44AA00AAA44400E244
8E002222A40"
3060 DATA "0084444000444480 04A04
44000000F004A8C8E000E2EAF088EAA
AE000E888E022EAAAE000EAE8E00064E
44400EAAE2E"
3070 DATA "88EAAAA00404446002022
22E88ACCAA0C44444E000AEEAA000CAA
AA000EAAAE000EAAE8802EAAE2200688
88000E842E0"
3080 DATA "44E444600 0AAAAF000AAA
A4000AAAEA000AA44A000AAAE2E00E24
8E024E2EAF024EAE8E0240444 6024EAF
AE024AAAAF0"
3090 DATA 4300
4000 REM BLOQUE 4
4005 DATA 65218,291
4010 DATA "2125FF110A5B010E00EDB
0CD007E2133FF110A5B011900EDB0CD0
07E214CFF110A5B012400"
4020 DATA "EDB0CD007E2170FF110A5
B012300EDB0CD007E2193FF110A5B011
700EDB0CD007E21AAFF11"
4030 DATA "0A5B011700EDB0CD007E2
1C1FF110A5B012400EDB0CD007EC9400
80203084F5043494F4E45"
4040 DATA "53FF1030010208312D455
34352494249522055 4E2054455854 4FF
F1040010208322D414C4D"
4050 DATA "4143454E4152204C41205
04147494E4120454E2043555253 4FFF1
0500102083 32D494D5052"
4060 DATA "494D4952204C412050414
7494E4120454E2043555253 4FFF10600
10208342D53414C564152"
4070 DATA "20454E2043494E5441FF1
070010208352D434152474152204445 2
043494E5441FF10800102"
4080 DATA "08362D4C45455220 4C415
320504147494E4153204 14C4D4143454
E41444153FF"
4090 DATA 21902
4100 DATA 0,0
4200 REM Este programa tarda dos
 minutos en ejecutarse PACIENCIA!
```

do extendido, desplazará el cursor hasta el siguiente tope de tabulación; para borrar los topes pulse «9» en modo extendido.

Margen adentrado

Si desea utilizar un margen izquierdo adentrado para un párrafo, coloque el cursor sobre el inicio del párrafo, pase a modo extendido y pulse «P»; ahora siga escribiendo normalmente; cada vez que pulse ENTER el cursor se situará en la columna donde estaba cuando pulsó «P», generando así un párrafo adentrado; para volver al margen izquierdo normal pase a modo extendido y pulse «B», que borrará el párrafo adentrado.

Para terminar la página pase a modo extendido y pulse «ENTER».

Si desea acentuar una vocal, púlsela en modo extendido, con lo que conseguirá la vocal acentuada.

Con el fin de aprovechar al máximo la memoria hemos omitido las sentencias REM, por lo que se adjunta un mapa de las distintas rutinas sobre el programa en BASIC, así como un mapa total de distribución de la memoria.

RUTINA PARA IMPRIMIR LA PAGINA CERO EN PANTALLA

NOMBRE: C.M. (1.ª parte).
ENTRADAS: Ninguna.
SALIDAS: Texto de Pág. cero a pantalla.
LONGITUD: 90 Bytes.
DIRECCION DE ARRANQUE: 32595 d.
DIRECCION DE INICIO: 32589 d.
TIEMPO DE EJECUCION: 1 seg.

Direcc.	Código Máquina	Listado		Assembler	
32589	62,13	ϕ1ϕ	LB7	LD A,ϕD	;Carga Código ENTER
32591	215	ϕ2ϕ		RST 1ϕ	;Haz nueva línea
32592	35	ϕ3ϕ		INC HL	;Incrementa puntero
32593	24,8	ϕ4ϕ		JR LB1	;Salta a LB1
32595	62,2	ϕ5ϕ		LD A,2	;Abre canal
32597	2ϕ5,1,22	ϕ6ϕ		CALL 16ϕ1	;número 2
326ϕϕ	33,194,133	ϕ7ϕ		LD HL,85C2	;Inicializa puntero
326ϕ3	62,194	ϕ8ϕ	LB1	LD A,194	;Direcciona
326ϕ5	5ϕ,54,92	ϕ9ϕ		LD (5C36),A	;primer
326ϕ8	62,126	1ϕϕ		LD A,126	;juego de
3261ϕ	5ϕ,55,92	11ϕ		LD (5C37),A	;caracteres
32613	126	12ϕ		LD A,(HL)	;Carga carácter
32614	254,255	13ϕ		CP FF	;Si es 255
32616	4ϕ,6ϕ	14ϕ		JR Z,LB6	;salta a LB6
32618	126	15ϕ		LD A,(HL)	;Carga carácter
32619	254,13	16ϕ		CP ϕD	;Si es 13
32621	4ϕ,222	17ϕ		JR Z,LB7	;salta a LB7
32623	126	18ϕ		LD A,(HL)	;Carga carácter
32624	246,ϕ	19ϕ		OR ϕ	;Si es cero
32626	4ϕ,42	2ϕϕ		JR Z,LB4	;salta a LB4
32628	215	21ϕ	LB2	RST 1ϕ	;Imprime carácter
32629	62,8	22ϕ		LD A,8	;Retrocede
32631	215	23ϕ		RST 1ϕ	;cursor
32632	35	24ϕ		INC HL	;Incrementa puntero
32633	62,129	25ϕ		LD A,129	;Direcciona segundo
32635	5ϕ,55,92	26ϕ		LD (5C37)	;juego de caracteres
32638	126	27ϕ		LD A,(HL)	;Carga carácter
32639	254,255	28ϕ		CP FF	;Si es 255
32641	4ϕ,35	29ϕ		JR Z,LB6	;salta a LB6
32643	126	3ϕϕ		LD A,(HL)	;Carga carácter
32644	254,13	31ϕ		CP ϕD	;Si es 13
32646	4ϕ,197	32ϕ		JR Z,LB7	;salta a LB7
32648	126	33ϕ		LD A,(HL)	;Carga carácter
32649	246,ϕ	34ϕ		OR ϕ	;Si es cero
32651	4ϕ,21	35ϕ		JR Z,LB5	;salta a LB5
32653	215	36ϕ	LB3	RST 1ϕ	;Imprime carácter
32654	35	37ϕ		INC HL	;Incrementa puntero
32655	229	38ϕ		PUSH HL	;Guarda puntero
32656	33,66,139	39ϕ		LD HL,8B42	;Carga tope puntero
32659	193	4ϕϕ		POP BC	;Carga puntero
3266ϕ	55	41ϕ		SCF	;Acarreo a uno
32661	63	42ϕ		CCF	;Acarrero a cero
32662	237,66	43ϕ		SBS HL,BC	;Si es fin de pág.
32664	4ϕ,12	44ϕ		JR Z,LB6	;salta a LB6
32666	197	45ϕ		PUSH BC	;Restituye
32667	225	46ϕ		POP HL	;puntero
32668	24,189	47ϕ		JR LB1	;Salta a LB1
3267ϕ	62,32	48ϕ	LB4	LD A,2ϕ	;Carga espacio
32672	24,21ϕ	49ϕ		JR LB2	;Salta a LB2
32674	62,32	5ϕϕ	LB5	LD A,2ϕ	;Carga espacio
32676	24,231	51ϕ		JR LB3	;Salta a LB3
32678	2ϕ1	52ϕ	LB6	RET	;Retorna

MAPA DE MEMORIA DEL EDITEXT

PAGINA 1 — 1.4ϕ8 — 35658
PAGINA EN CURSO — 1.4ϕ8 — 34242
SEGUNDO JUEGO DE CARACTERES — 768 — 33474
PRIMER JUEGO DE CARACTERES — 768 — 327ϕ6
RUTINAS DE TRANSFERENCIA — 15 — 32691
IMPRESION PAGINA ϕ — 12 — 32679
90 — 32589
56
RUTINA DE ROTULOS — 277 — 32256
RAMTOP — 32255
AREA BASIC — 15.872 — 16384
ROM — 16 K — 0
PAGINA 9 — 1.4ϕ8 — 46914
PAGINA 8 — 1.4ϕ8 — 455ϕ6
PAGINA 7 — 1.4ϕ8 — 44ϕ98
PAGINA 6 — 1.4ϕ8 — 4269ϕ
PAGINA 5 — 1.4ϕ8 — 41282
PAGINA 4 — 1.4ϕ8 — 39874
PAGINA 3 — 1.4ϕ8 — 38466
PAGINA 2 — 1.4ϕ8 — 37ϕ58
PAGINA 17 — 1.4ϕ8 — 58178
PAGINA 16 — 1.4ϕ8 — 5677ϕ
PAGINA 15 — 1.4ϕ8 — 55362
PAGINA 14 — 1.4ϕ8 — 53954
PAGINA 13 — 1.4ϕ8 — 52546
PAGINA 12 — 1.4ϕ8 — 51138
PAGINA 11 — 1.4ϕ8 — 4973ϕ
PAGINA 1ϕ — 1.4ϕ8 — 48322
P-RAMT — 26 — 65535
RUTINA "Cmp" — 292 — 65218
PAGINA 21 — 1.4ϕ8 — 6381ϕ
PAGINA 20 — 1.4ϕ8 — 624ϕ2
PAGINA 19 — 1.4ϕ8 — 6ϕ994
PAGINA 18 — 1.4ϕ8 — 59586

Mapa de memoria del EDITEXT.

RUTINA PARA TRANSFERIR LA PAGINA CERO A CUALQUIER PAGINA DE ALMACENAMIENTO

NOMBRE: C.M. (2.ª parte)
ENTRADAS: En las direcciones 3268ϕ y 32681 se introduce la dirección de la página de destino (cuando se carga, estos valores quedan inicializados para tomar como destino la página uno).
SALIDAS: La página cero es transferida a la página de destino.
LONGITUD: 12 Bytes
DIRECCION DE ARRANQUE: 32679
DIRECCION DE INICIO: 32679
TIEMPO DE EJECUCION: 2ϕ miliseg.

Direcc.	Código Máquina	Listado	Assembler	
32679	17,66,139	ϕ1ϕ	LD DE,8B42	;Carga destino
32682	33,194,133	ϕ2ϕ	LD HL,85C2	;Carga origen
32685	1,128,5	ϕ3ϕ	LD BC,ϕ58ϕ	;Carga longitud
32688	237,176	ϕ4ϕ	LDIR	;Transfiere incrementa y repite
3269ϕ	2ϕ1	ϕ5ϕ	RET	;Retorna y repite

CAMPOS DE ENERGIA

Francisco LORCA SALAS

Spectrum 48 K

NOTAS GRAFICAS

A B C D E F G H I J

Para ubicarnos en una central nuclear no hace falta contar con una gran imaginación aunque, en este programa que veremos a continuación, se encuentre en cualquier espacio interestelar.

Así pues, situémonos en esa central nuclear en donde nosotros seremos el guarda. Nuestra misión será la de procurar que los pequeños escapes que se producen, no salgan de las cuatro torres representadas en la pantalla. Si no lo conseguimos, iremos perdiendo un poco de energía con cada escape y, si éste es muy grande, perderemos una vida.

A pesar de que disponemos de siete vidas, las iremos perdiendo, también, cada vez que pulsemos fuego (0) fuera de los extremos rectangulares de la torre o si, en ese momento, no hay ningún escape en ella.

La partida finaliza cuando se pierden las siete vidas de que disponemos. Para jugar, existen tres controles: arriba; «1» ó «7»; abajo: «2» ó «6»; fuego: «0».

```
   1 OVER 0: BORDER 0: PAPER 0:
INK 7: CLS
   2 REM F
   3 LET q$="?????"
   5 GO SUB 9800
   7 GO SUB 9840
   8 LET h=0
  10 GO SUB 8000
  20 GO SUB 9000
  25 PRINT AT 0,23;q$;": ";h
  30 OVER 1
  35 PRINT INK 3;AT y,x;"[gráfico]"
  39 IF li<=1 THEN GO TO 7000
  40 LET p=p-.05: PRINT OVER 0;A
T 0,0;s: LET g=1: GO TO (INT (RN
D*4)+1)*100
  60 FOR d=1 TO p: PRINT INK 7;A
T y,x;"[gráfico]"
  65 IF INKEY$="1" OR INKEY$="7"
 THEN LET x=x+1: LET y=y-1
  70 IF INKEY$="2" OR INKEY$="6"
 THEN LET x=x+1: LET y=y+1
  75 IF y=-1 THEN LET y=21
  80 IF y=22 THEN LET y=1
  83 IF x=-1 THEN LET x=30
  87 IF x=31 THEN LET x=0
  90 PRINT INK 3;AT y,x;"[gráfico]"
  92 IF INKEY$="0" THEN IF g THE
N GO SUB 1000
  93 NEXT d
  95 RETURN
 100 LET o=s: FOR f=1 TO 4
 105 LET a=1
 110 PRINT AT 10+2*f,15; INK 5;a
$(f)
 115 GO SUB 60
 120 BEEP .04,f
 130 PRINT AT 10+2*f,10;a$(f);AT
11+2*f,15; INK 5;a$(f)
 135 GO SUB 60
 140 BEEP .04 ,f
 150 PRINT AT 11+2*f,15;a$(f)
 160 NEXT f
 165 IF s<>o THEN PRINT FLASH 1;
AT 11+2*f,15; INK 6;"[gráfico]": BEEP .1
,-30: BEEP .1,-28: BEEP .1,-26:
BEEP .1,-28: BEEP .1,-30: PRINT
AT 11+2*f,15; INK 7;"[gráfico]"
 167 IF s=o THEN LET li=li-.2: P
RINT AT 21,21;"VIDAS";AT 21,27;
OVER 0; INT li;" "
 170 GO TO 39
 200 LET o=s: FOR f=1 TO 4
 205 LET a=2
 210 PRINT AT 11-2*f,16; INK 3;a
$(f)
 215 GO SUB 60
 220 BEEP .04,f
 230 PRINT AT 11-2*f,16;a$(f);AT
10-2*f,16; INK 3;a$(f)
 235 GO SUB 60
 240 BEEP .04,f
 250 PRINT AT 10-2*f,16;a$(f)
 260 NEXT f
 265 IF s<>o THEN PRINT FLASH 1;
AT 10-2*f,16; INK 6;"[gráfico]": BEEP .1
?-30: BEEP .1,-28: BEEP .1,-26:
BEEP .1,-28: BEEP .1,-30: PRINT
AT 10-2*f,16; INK 7;"[gráfico]"
 267 IF s=o THEN LET li=li-.2: P
RINT AT 21,27; OVER 0;INT li;"
";AT 21,21; OVER 0;"VIDAS"
 270 GO TO 39
 300 LET o=s: FOR f=1 TO 5
 305 LET a=3
 310 PRINT AT 10,15-2*f; INK 4;b
$(f)
 315 GO SUB 60
 320 BEEP .04,f*6
 330 PRINT AT 10,15-2*f;b$(f);AT
10,14-2*f; INK 4;b$(f)
 335 GO SUB 60
 340 BEEP .04,f*6
 350 PRINT AT 10,14-2*f,b$(f)
 360 NEXT f
 365 IF s<>o THEN PRINT FLASH 1;
AT 10,14-2*f; INK 6;"[gráfico]": BEEP .1
,-30: BEEP .1,-28: BEEP .1,-26:
```



```
BEEP .1,-28: BEEP .1,-30: PRINT
AT 10,14-2*f; INK 7;"▒"
 367 IF s=o THEN LET li=li-.2: P
RINT AT 21,27; OVER 0;INT li;AT
21,21; OVER 0;"VIDAS"
 370 GO TO 39
 400 LET o=s: FOR f=1 TO 5
 405 LET a=4
 410 PRINT AT 11,16+2*f; INK 6;b
$(f)
 415 GO SUB 60
 420 BEEP .04,f*6
 430 PRINT AT 11,16+2*f;b$(f);AT
11,17+2*f; INK 6;b$(f)
 435 GO SUB 60
 440 BEEP .04,f*6
 450 PRINT AT 11,17+2*f;b$(f)
 460 NEXT f
 465 IF s<>o THEN PRINT FLASH 1;
AT 11,17+2*f; INK 6;"▒": BEEP .1
,-30: BEEP .1,-28: BEEP .1,-26:
BEEP .1,-28: BEEP .1,-30: PRINT
AT 11,17+2*f; INK 7;"▒"
 467 IF s=o THEN LET li=li-.2: P
RINT AT 21,27; OVER 0;INT li;AT
21,21; OVER 0;"VIDAS"
 470 GO TO 39
1000 IF x<6 THEN GO TO 1100
1010 IF x>9 AND x<22 THEN GO TO
1200
1020 IF NOT x>24 THEN GO TO 2000
1030 IF x>29 OR y<2 OR y>19 THEN
 GO TO 2000
1040 FOR q=2 TO 19: PRINT PAPER
3;AT q,27;"  ": NEXT q
1050 BEEP .03,-10: BEEP .03,-13:
 BEEP .03,-15: BEEP .03,-10
1060 FOR q=2 TO 19: PRINT PAPER
0;AT q,27;"  ": NEXT q
1065 IF a<>4 THEN GO TO 2000
1070 LET s=s+20
1080 LET g=0
1090 RETURN
1100 IF x<1 OR y<2 OR y>19 THEN
GO TO 2000
1110 FOR q=2 TO 19: PRINT PAPER
3;AT q,2;"  ": NEXT q
1120 BEEP .03,-10: BEEP .03,-13:
 BEEP .03,-15: BEEP .03,-10
1130 FOR q=2 TO 19: PRINT PAPER
0;AT q,2;"  ": NEXT q
1135 IF a<>3 THEN GO TO 2000
1140 LET s=s+20
1150 LET g=0
1160 RETURN
1200 IF y>10 THEN GO TO 1300
1210 IF y<1 OR y>3 THEN GO TO 20
00
1220 FOR q=11 TO 20: PRINT PAPER
 3;AT 1,q;" ";AT 2,q;" ";AT 3,q;
" ": NEXT q
1230 BEEP .03,-10: BEEP .03,-13:
 BEEP .03,-15: BEEP .03,-10
1240 FOR q=11 TO 20: PRINT PAPER
 0;AT 1,q;" ";AT 2,q;" ";AT 3,q;
" ": NEXT q
1245 IF a<>2 THEN GO TO 2000
1250 LET s=s+20
1260 LET g=0
1270 RETURN
1300 IF y>20 OR y<18 THEN GO TO
2000
1310 FOR q=11 TO 20: PRINT PAPER
 3;AT 18,q;" ";AT 19,q;" ";AT 20
,q;" ": NEXT q
1320 BEEP .03,-10: BEEP .03,-13:
 BEEP .03,-15: BEEP .03,-10
1330 FOR q=11 TO 20: PRINT PAPER
 0;AT 18,q;" ";AT 19,q;" ";AT 20
,q;" ": NEXT q
1335 IF a<>1 THEN GO TO 2000
1340 LET s=s+20
1350 LET g=0
1360 RETURN
2000 PRINT AT y,x;"▶◀"
2010 FOR r=y TO 21
2020 PRINT AT r,x-1; INK 5;"▒▒▒"
2030 BEEP .001,r
2040 PRINT AT r,x-1;"▒▒▒"
2050 NEXT r
2060 LET li=li-1: PRINT OVER 0;A
T 21,27;INT li;"  ": IF li<=1 TH
EN GO TO 7000
2065 PRINT AT y,x;"▶◀"
2070 RETURN
7000 FOR f=1 TO 60: PRINT OVER 0
;AT 10,8; INK f/9; INVERSE 1;"G
A M E  O V E R": BEEP .001,f: NE
XT f
7005 IF s>h THEN LET h=s: INPUT
"RECORD:INTRODUCE TU NOMBRE
 (4 Letras maximo)", LINE q$
7100 PRINT OVER 0;AT 18,4; INVER
SE 1;"PULSE CUALQUIER TECLA";AT
20,5; INVERSE 1;"PARA VOLVER ▪ E
MPEZAR";AT 21,0; INVERSE 0; INK
6;"RECORD    ";q$
7200 IF INKEY$<>"" THEN RESTORE
: CLS : OVER 1: GO TO 10
7300 GO TO 7200
8000 FOR f=USR "a" TO USR "j"+7
8010 READ x: POKE f,x: NEXT f
8020 DATA 40,BIN 01010100,BIN 10
111010,124,BIN 10111010,BIN 0101
0100,40,0
8030 DATA 0,16,40,BIN 01010100,4
0,16,0,0
8040 DATA 0,0,0,24,24,0,0,0
8050 DATA 0,0,0,16,0,0,0,0
8060 DATA 255,195,189,189,189,18
9,195,255
8070 DATA 0,62,54,42,54,62,0,0
8080 DATA 0,0,28,28,28,0,0,0
8090 DATA 96,240,249,239,249,240
,96,0
8100 DATA 6,15,159,247,159,15,6,
0
8110 DATA BIN 11011010,33,76,195
,24,BIN 01100101,BIN 10010011,BI
N 01100100
8250 LET x=1: LET y=1
8300 LET a$="··◆✱"
8350 LET b$="··▪■▩"
8400 LET s=0
8410 LET li=8: LET p=5
8500 RETURN
9000 OVER 0: PLOT 126,93: DRAW 4
,0: DRAW 0,-4: DRAW 4,0: DRAW 0,
-4: DRAW -4,0: DRAW 0,-4: DRAW -
4,0: DRAW 0,4: DRAW -4,0: DRAW 0
,4: DRAW 4,0: DRAW 0,4
9010 DRAW -40,77: DRAW 84,0: DRA
W 0,-25: DRAW -84,0: DRAW 0,25:
PLOT 86,145: DRAW 40,-52: DRAW 4
,0
9020 DRAW 40,52: DRAW 0,25: DRAW
 -40,-77
9030 PLOT 126,81: DRAW -40,-77:
DRAW 84,0: DRAW 0,25: DRAW -84,0
: DRAW 0,-25: DRAW 0,25: DRAW 40
,52: DRAW 4,0: DRAW 40,-52: DRAW
 0,-25: DRAW -40,77
9040 PLOT 122,89: DRAW -107,71:
DRAW 30,0: DRAW 0,-146: DRAW -30
,0: DRAW 0,146: DRAW 30,0: DRAW
77,-71: DRAW 0,-4: DRAW -107,-71
: DRAW 30,0: DRAW 77,71
9050 PLOT 134,89: DRAW 107,71: D
RAW -30,0: DRAW 0,-146: DRAW 30,
0: DRAW 0,146: DRAW -30,0: DRAW
-77,-71: DRAW 0,-4: DRAW 107,-71
: DRAW -30,0: DRAW -77,71
9500 RETURN
9800 FOR f=1 TO 7: FOR g=1 TO f:
 PRINT INK g;AT f,f*2;"CAMPOS DE
 ENERGIA": BEEP .03,g: NEXT g: N
EXT f
9835 RETURN
9840 PRINT AT 10,0;"Defiende el
tubo con tu nave    construyendo
 campos de energia   al final de
 los cuatro finales abiertos del
 sistema.Si intentasponer los ca
mpos en otro sitio  que no sean
los finales  resul-  taras destr
uido"
9850 PRINT "Si pones un campo de
 energia sinenemigo dentro,seras
 destruido": PRINT AT 21,4;"Por
Francisco Lorca.": PAUSE 150: PR
INT AT 21,4;"    Pulsa una tecla
"
9855 PAUSE 0: CLS : GO SUB 9800
9860 PRINT AT 9,0;"Mueve con:1(a
rriba)y 2(abajo),  pero recuerda
 que te mueves pa- ra la derecha
 al mismo tiempo y si te sales p
or un lado de la   pantalla lleg
as al otro."
9870 PRINT "Instale un campo de
energia en  la boca de un tubo c
on (0).    Si dejas pasar un en
emigo o te  autodestruyes pierde
s energia, si la perdida es muy
 grande el  juego termina."; IN
VERSE 1; OVER 1;"Pulse una tecla
": PAUSE 0: CLS
9880 RETURN
```

García Gete hace balance de la campaña navideña

INVESTRONICA, VIENTO EN POPA

Gabriel NIETO

Tras la fuerte campaña de ventas navideñas nos pusimos en contacto con el Director de Investrónica, Ricardo García Gete, para que nos contara cómo habían ido las ventas más fuertes del año.

En la sede de Investrónica, en una especie de pequeña habitación que parecía estar pensada expresamente para recibir visitas, se encontraba Ignacio Argote y Ricardo García Gete. Con este último, mantuvimos una larga conversación. Papeles en mano y esperando mi primera e inevitable pregunta, las ventas navideñas, comenzó la entrevista.

«Nosotros nos habíamos planteado una serie de objetivos de cara a esta campaña, y en base a esto habíamos estructurado los stocks y la publicidad. Nuestra idea de ventas estaba en torno a los 40.000 ordenadores y la realidad es que, no sólo se han cumplido las previsiones, sino que además se han superado. Hemos vendido alrededor de 50.000.»

Para ser exactos, han sido 52.000, de los cuales se puede efectuar el siguiente desglose:

—35.000 Spectrum.
—15.000 Plus.
— 2.000 QL.

Estas cifras sobrepasan, aproximadamente en un 30%, las previsiones que tenían hechas para la campaña. A mediados de diciembre y, debido al crecimiento experimentado durante la primera quincena de la demanda, Investrónica tiene que pedir 15.000 ordenadores más.

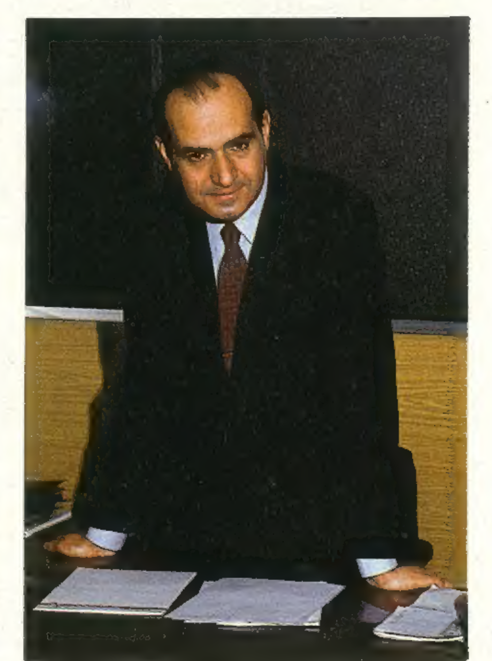
«Superando previsiones, hemos alcanzado una venta de 50.000 ordenadores durante las Navidades.»

Cuando le preguntamos por la acogida del Plus a García Gete, su respuesta fue rotunda. *«Muy buena»*; tras lo cual matizó: *«Quizá ha entrado un poco tarde y esto ha motivado que la gente que no lo conocía siguiera pidiendo Spectrum. De todas formas, ha sido muy buena ya que, pese a ser un producto que se introdujo a primeros de noviembre, lo presentamos en el SIMO como primicia mundial en una feria y, a finales de noviembre, principios de diciembre, llegó a los distribuidores.»*

El prelazamiento del «QL»

A pesar de todo, el rey sigue siendo el Spectrum: *«Si hubiéramos tenido más, hubiéramos vendido más.»*

Cuando hablamos del QL, tema que por otra parte no podía faltar en nuestra conversación, nuestro entrevistado nos aclaró algunos aspectos que parece ser, no estaban suficientemente claros. *«El lanzamiento que hemos hecho del QL no ha sido el lanzamiento oficial que va a hacer Investrónica, es decir, cuando lo hagamos se hará con las condiciones habituales de nuestros lanzamientos, toda la documentación en castellano y todos los programas también. Esto ha sido un prelanzamiento del producto, motivado por la demanda que había en el mercado.»*



Esto parece ser que no ha sido impedimento para que, en menos de un mes, ya se hayan vendido 2.000 QL, cifra muy estimativa si tenemos en cuenta el tipo de ordenador que es.

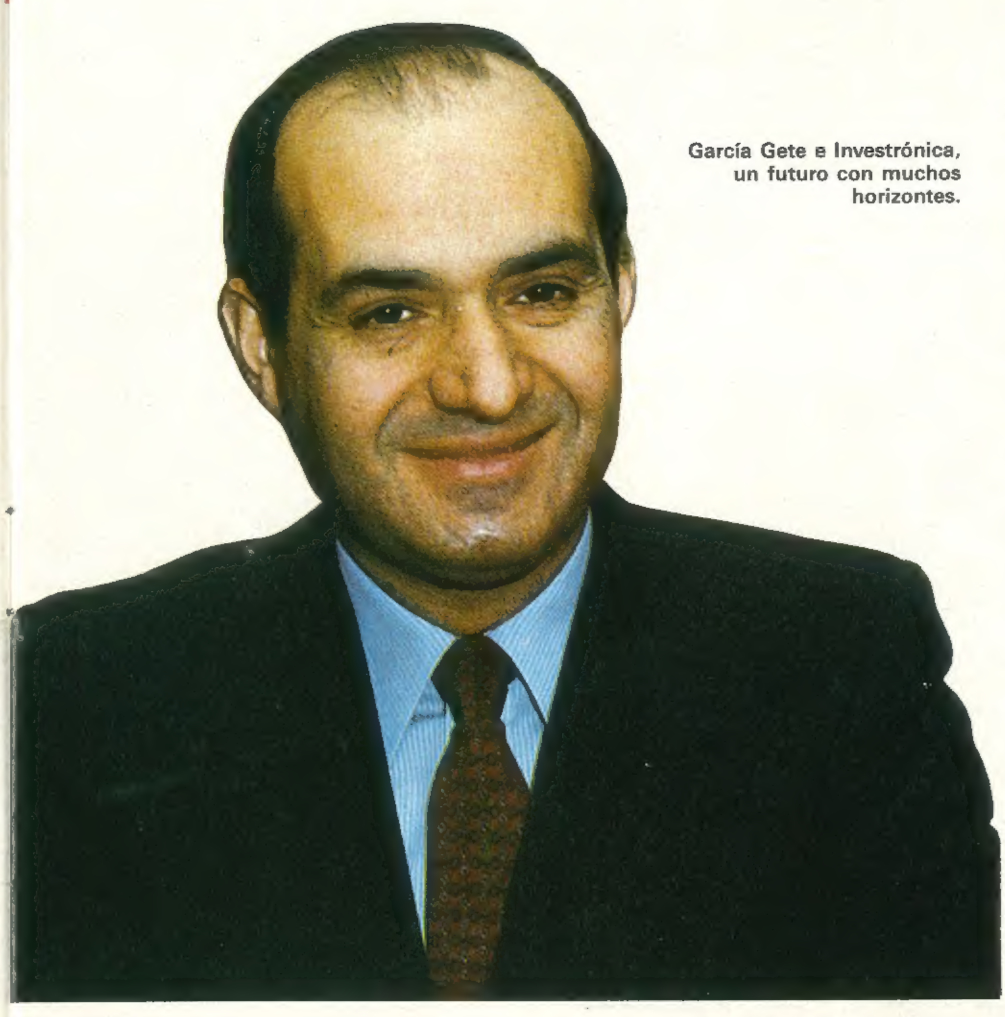
El lanzamiento definitivo se hará sobre el 15 de febrero, aunque depende de que Sinclair termine la versión definitiva en inglés. Ya está hecho el teclado en castellano y la versión de los cuatro programas, sólo falta que a los últimos detalles del sistema operativo les den el visto bueno. De todas formas, el interés por este ordenador se ha hecho patente en las oficinas de Investrónica.

Hay un dato estimativo en toda esta campaña, que ha seguido marcando el rumbo de los ordenadores domésticos: la supremacía del Spectrum, incluso por encima de la nueva versión. *«La gente conoce el Spectrum, entonces cuando le ofreces algo que sea distinto de Spectrum, aunque como en este caso sea también Spectrum, la gente tiene sus dudas, sobre todo el que no sabe nada, no quiere otra cosa más que lo que ha visto a su vecino. También ha influido mucho el precio y los ocho paquetes de regalo.»*

Los tres factores han influido decisivamente: desconocimiento, precio y las cintas; a pesar de todo ello, las previsiones son buenas para un futuro y parece ser que el Plus será una alternativa del Spectrum, pero que todavía está muy lejos de desbancar a éste totalmente del mercado. El teclado, que es la única diferencia, parece ser que es aceptado por muchos compradores, sobre todo por los más jóvenes que no les preocupa y además se sienten mucho más a gusto con él.

Los discos del Spectrum

Otra pregunta que estaba en el aire, era la de los traídos y llevados discos para el Spectrum, los cuales, a pesar de haber estado presentes en el SIMO, siguen sin aparecer, lo que puede suponer, a su vez, un hándicap para el Spectrum, frente a otros ordenadores que ya salen al mercado directamente incorporando esta opción. *«La unidad que nosotros hemos seleccionado ha sido creada por Timex, una casa muy fiable en el campo del Spectrum. El motivo por el que no ha salido antes, es porque hemos querido dejar pasar primero la campaña de Navidad para volcarnos durante el año 85 en el lanzamiento de este tipo de periféricos.»*



García Gete e Investrónica, un futuro con muchos horizontes.

Los MSX era un tema que, por supuesto, no podía faltar en esta entrevista; por eso, intentamos abordarlo aunque ya nos imaginábamos la respuesta. *«No, no nos preocupa y además por el momento, tampoco está preocupando en Europa, ni en Estados Unidos. En este último país el líder del mercado no es MSX, como tampoco lo es en Europa. Por la información que nosotros tenemos ni a Sinclair, ni a Commodore les preocupa MSX, al menos por el momento. Hay que tener en cuenta que los dos líderes están perfectamente consolidados en el mundo que no sea japonés. Va a ser muy difícil su penetración en los demás mercados porque además hay ya otros productos que no son MSX.»*

El mercado español y el inglés, según él, siguen cierto paralelismo, por eso, nos hacía hincapié en que en ninguno de los países se ha conseguido el éxito, que en principio se esperaba, de los colosos nipones.

A nivel de software, Investrónica sigue mostrándose conservadora y ha adoptado una línea muy parecida de actuación en este campo a la de Sinclair. *«Nosotros no tenemos más títulos que tiene Sinclair Research, lo que ellos sí hacen es que seleccionan solamente aquellos programas que consideran que son los más idóneos y los de mayor difusión. Sin embargo, lo que sí hacen es fomentar que existan gran cantidad de compañías que vendan software. En Investrónica se hace lo mismo, preferimos que los demás creen software para Spectrum y en ese sentido colaboramos enormemente.»*

Para Investrónica, sin embargo, la piratería de software sigue siendo un problema también incluso para una empresa tan poderosa como ésta. La administración parece ser que, de momento, está preocupada, pero hasta la fecha no ha pasado de ser sólo una preocupación. *«Es un problema muy grave que además puede limitar mucho la creación de nuevos programas. Vender, en este país, pocos millares de copias es todo un récord, cuando lo normal para el parque de Spectrum existentes sería vender muchos miles.»* Todo esto no influye, sin embargo, para que Investrónica siga tratando de apoyar la creación de un mercado importante de software.

Software para el «QL»

Lo que parece un hecho es que el QL va a traer una nueva serie de programas de aplicación que lo van a diferenciar bastante del Spectrum. *«El software del QL va a ser bastante distinto, la mayoría de los programas van a ser de aplicación, lo que sí se está tratando de hacer en Inglaterra es algún tipo de emulador de Spectrum, de tal manera que el software de éstos pudiera valer para correr en QL. Esto es algo a nivel de rumor. De todas formas el software del QL va a ser más limitado ya que va a ir dirigido a un sector más profesional aunque sea a nivel particular.»*

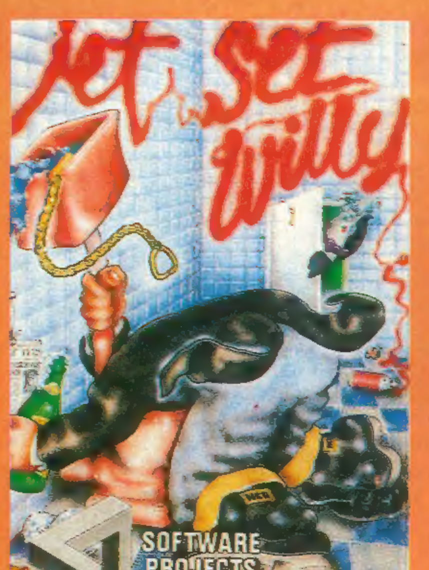
El QL, lo que no cabe duda es de que va dirigido a un público muy determinado, que se va a comprar un ordenador para darle una aplicación concreta, incluso en algunos casos para una única aplicación, como pueda ser, por ejemplo, para llevar un tratamiento de textos.

El número de ordenadores en la actualidad, tras la fuerte campaña de ventas navideñas, se encuentra entre 175.000 y 200.000 ordenadores, lo que convierte al Spectrum en un líder consolidado. Por eso preocupa a los creadores del Spectrum cuál va a ser la tendencia de ese potencial futuro comprador de un ordenador con mayores prestaciones. *«Es evidente que casi todos los usuarios que en su momento tuvieron el ZX 81, aproximadamente en un 80%, a muchos de ellos para las necesidades informáticas que tienen, les servirá con esto; pero otros muchos van a pasar al QL. Lo que no podemos cuantificar, es cuántos van a pasar porque la realidad depende de muchos factores.»*

En un futuro no muy lejano, el QL contará con compiladores para otros lenguajes profesionales.

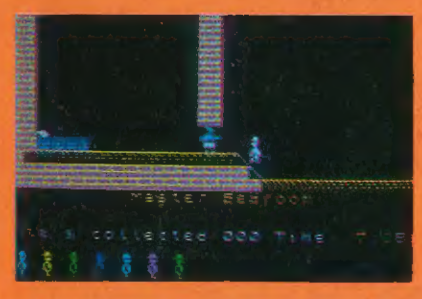
Las esperanzas de futuro, está claro que están puestas en este nuevo ordenador que además, intentará ampliar su campo y sus prestaciones. *«Una de las cosas que va a tener el QL va a ser compiladores para otra serie de lenguajes, en estos momentos se está hablando de Pascal, Cobol y algún otro tipo de lenguaje.»*

La entrevista terminaba y la conclusión era clara: Spectrum sigue siendo, con diferencia, el rey del mercado del ordenador doméstico, e Investrónica va a continuar su campaña, una campaña, por cierto, plagada de éxitos.











Desventuras de un nuevo rico

JET SET WILLY

Software Projects/Ventamatic
48 K
Tipo de juego: Arcade
PVP: 1.700

Es la segunda parte del Manic Miner, uno de los juegos más famosos de entre los que han sido realizados para el Spectrum.

Esta continuación nos traslada al maravilloso y peligroso mundo de Willy, el personaje que veíamos en la primera parte. El minero, tras lograr llegar al final de su aventura, ha conseguido hacerse rico y vive en una gran mansión. Su objetivo y única obsesión es la de entrar en su habitación a dormir, pero su ama de llaves se lo impide, porque toda la casa está llena de botellas y desperdicios tras una ruidosa fiesta. Willy tendrá que lograr recoger todas éstas antes de poder entrar en la habitación.

La casa está llena de extraños lugares que suponen una continua trampa para Willy, cuchillas asesinas, bolas de fuego, sierras que salen del suelo, monaguillos que nos atacan, tazas del báter que se mueven solas y una larga lista de peligros que nos acechan continuamente.

La mecánica del juego es muy similar a la del Manic Miner, e incluso alguno de los objetos son iguales, pero con la diferencia de que en esta ocasión, además de haber muchos más, son también más originales.

El juego tiene muchas pantallas y todas ellas están construidas en forma laberíntica, se comunican por todos los lados. En esto, también es diferente a su antecesor que se limitaba a pasar de una pantalla a otra. Aquí se puede retroceder, subir, bajar, ir a la derecha o a la izquierda, y por supuesto, regresar a un lugar donde ya se ha estado con anterioridad.

Los gráficos están muy bien construidos y la distribución de la pantalla está muy bien pensada, todos los objetos se mueven de forma rítmica y acompasada, y cada trampa está pensada para dificultarnos todo lo posible nuestra labor. En la mayoría de los casos es necesario estudiar con anterioridad todos los movimientos que vamos a efectuar.

Muchas de las habitaciones por las que pasamos están comunicadas por medio de escaleras, para lograr pasarlas será necesario que saltemos sobre ellas, pero al hacerlo, hay que tener mucho ciudado, porque si lo hacemos desde demasiado alto no podremos conseguirlo.

El juego es tremendamente original y, aunque está basado en muchas cosas en el anterior, reúne los suficientes alicientes para hacer deshonor a la frase aquella de que «segundas partes nunca fueron buenas». En este caso, podemos asegurarles todo lo contrario. Es, incluso, mejor.

El nivel de dificultad es muy alto, por lo que podemos adelantar a todos los que jueguen con él, un buen nivel de adicción y la seguridad de que van a pasar un rato muy entretenido.











Destruye a los robots

SPECTRON

Virgin/Compulogical
48 K
Tipo de juego: Arcade
PVP: 1.350



Hay un tipo de juegos, que se caracterizan por su simplicidad creativa, que se hicieron muy famosos tiempo atrás. Se trata de aquellos en los que se nos presenta una pantalla con una serie de obstáculos y hay que conseguir, dentro de ella, destruir a un determinado número de enemigos, una vez que se ha logrado, podemos pasar a una pantalla nueva, que en realidad es la misma con la única diferencia de que los obtáculos son mayores y el nivel de dificultad aumenta. Este juego pertenece a ese tipo.

Tenemos que luchar contra un grupo de robots que intentarán acorralarnos y destruirnos. Para ello disponemos de una pistola láser con la que podremos acabar con ellos si acertamos en el blanco. Cada pantalla va aumentando el número de enemigos hasta llegar a la cifra de 54 robots que recorren implacablemente ésta, tratando de acorralarnos.

Además de los enemigos a los que nos enfrentamos, hay también un determinado número de vallas electrificadas que no podemos tocar, las cuales aumentarán progresivamente a medida que evolucione el juego.

Hay nueve niveles de dificultad, siendo los últimos muy difíciles de pasar.

El juego nos informa, periódicamente, de los puntos obtenidos en base al tiempo de permanencia en el juego, porcentaje de eficacia de nuestros disparos y puntos, dependiendo estos últimos de los enemigos destruidos. Estos son los Swarnmers, Speeders, Launchers y Electron. Cada uno tiene unas peculiaridades concretas que tenemos que tener en cuenta a la hora de enfrentarnos a ellos.

Los gráficos son muy simples, el movimiento está muy bien y el sonido es francamente bueno. Utilizado con un amplificador puede producir efectos espectaculares.

Un juego sin complicaciones en el que son necesarios muchos reflejos.

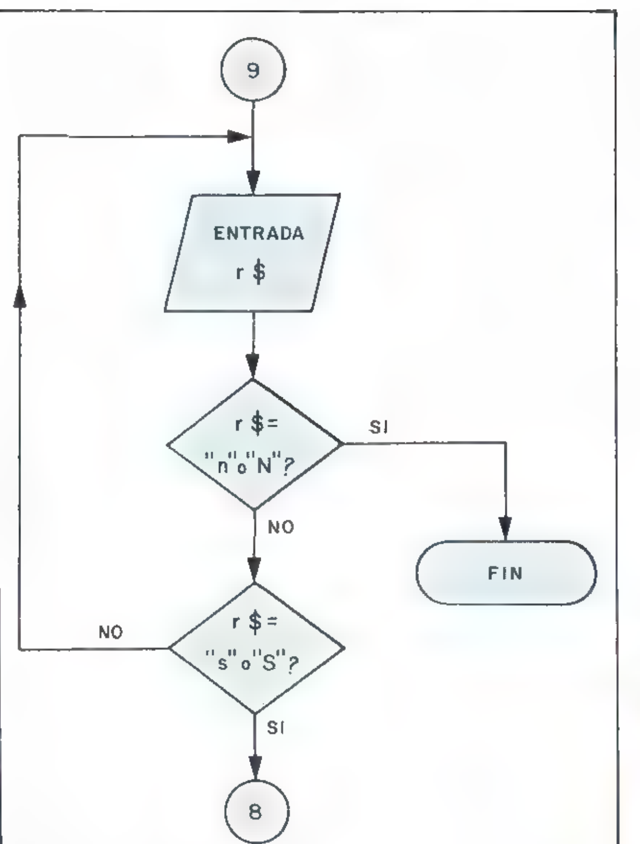
Procesando textos

CONTEXT V.6

Ventamatic
48 K
Tipo: Gestión
PVP: 4.000

Este programa es una nueva versión del popular procesador de textos, Context.

En esta ocasión, se ha incluido la posibilidad de utilizar, tanto el cassette



Programa "Areas" Rutina "Fin".

teriormente, por parejas hasta que sólo quede una condición, si el resultado es «ϕ» el ordenador pasará a ejecutar la siguiente instrucción; si es «1», ejecutará antes las instrucciones contenidas en el «THEN...».

Ejemplo:

IF (a > b OR c = 7) AND (t < > ϕ AND p = a * 5) THEN ...

para los valores:

a = 4
b = 2
c = 6
t = 5ϕ
p = 2ϕ

Resolviendo por pasos:

a) (a > b) OR (c = 7)
(a > b) es verdadero (1), ya que «4» es mayor que «2».
(c = 7) es falso (ϕ), ya que «6» no es igual a «7».

Realizando la operación lógica «OR» de los valores anteriores

1 OR ϕ

observamos que el resultado es «1», ya que se cumple una de las condiciones. (Repasar el capítulo dedicado a «OPERADORES LOGICOS»).

b) (t < > ϕ) AND (p = a * 5)
(t < > ϕ) es verdadero (1), ya que «5ϕ» es distinto de «ϕ».
(p = a * 5) es verdadero (1), por que «2ϕ» es igual a «4» por «5».
Realizando la operación lógica «AND» de los valores anteriores «1» y «1»

1 AND 1

tenemos un resultado igual a «1» ya que se cumplen las dos condiciones.

c) (condición 1) AND (condición 2)
Como las dos condiciones se cumplen, el resultado global es también verdadero «1».

Asigne otros valores a las variables e intente resolver el resultado; si tiene algún problema o desea comparar los resultados ejecute el siguiente programa:

```
10 REM ********************
       *                  *
       * VERDADERO/FALSO  *
       *                  *
       ********************
20 INPUT "A > ";a
30 INPUT "B > ";b
40 INPUT "C > ";c
50 INPUT "T > ";t
60 INPUT "P > ";p
70 IF (a>b OR c=7) AND (t<>0
AND p=a*5) THEN PRINT "VERDADERO" : GO TO 20
80 PRINT "FALSO"
90 GO TO 20
```

El valor de una condición también puede ser asignado a una variable, de la forma:

LET resultado = f$ = "FIN"

la variable «resultado» tendrá el valor «1» cuando f$ sea igual a la cadena «FIN», y «ϕ» cuando f$ tenga otro valor. Podríamos, por tanto, editar una sentencia del tipo:

IF resultado THEN ...

NEW

Acceso al teclado



MODO K

Definición

Generalmente este comando se utiliza de forma directa y no precisa de ningún argumento para poderse ejecutar.

«NEW» borra el programa o programas almacenados en memoria, también borra el valor de las variables definidas. Hay una serie de *variables de sistema* que no se ven afectadas por este comando, entre ellas los GDU o gráficos definidos por el usuario.

Debe utilizarse con mucho cuidado ya que de lo contrario, podríamos borrar un programa que aún no ha sido salvado, cosa no muy agradable por cierto.

Cuando se ejecuta da la impresión de haber conectado el aparato de nuevo, ya que nos presenta el famoso mensaje inicial:

© 1982 Sinclair Research Ltd.

Puede incluirse con precaución dentro de un programa para dar por finalizada su ejecución y borrado.

El programa n.º «2» incorpora esta sentencia, "sálvelo antes de ejecutar".

CLS

Acceso al teclado



MODO K

Definición

El comando «CLS» puede ser utilizado tanto en modo directo, como formando parte de un programa; no precisa de ningún argumento.

La función de este comando es borrar la pantalla de caracteres y gráficos, asumiendo ésta el color especificado en la última sentencia «PAPER» ejecutada con anterioridad. El color del *borde* de la pantalla no se ve afectado por esta sentencia.

Ejemplo de listado.

Ejemplos:

— Introduzca el siguiente programa:

```
10 REM *****************
       *               *
       * !!! @@@ !!!   *
       *               *
       *****************
20 FOR i=1 TO 704
30 PRINT "@";
40 NEXT i
50 REM
       *****************
       *               *
       * BORDE (ROJO)  *
       *               *
       *****************
60 BORDER ■
```

este programa llena la zona de visualización con el símbolo «@», teclee el comando directo «CLS» y observe la pantalla.

— En el siguiente programa la sentencia «CLS» se utiliza para borrar la pantalla y asumir los colores «magenta» para el fondo y «amarillo» el de los caracteres; el color del borde se asigna directamente con la sentencia «BORDER».



```
10 REM **********************
       *                    *
       * ASIGNAR COLORES    *
       *                    *
       **********************
20 REM ** BORDE "verde" **
30 BORDER 4
40 REM ** FONDO "magenta" **
50 PAPER 3
60 REM ** TINTA "amarilla" **
70 INK 6
80 CLS
90 PRINT AT 10,10;"MICROHOBBY"
```

Acceso al teclado

MODO K

Definición

«LIST» se utiliza normalmente como comando directo y permite obtener un listado del programa almacenado en memoria. La estructura general de esta sentencia es:

SENTENCIA	ARGUMENTO
LIST	N.º DE LINEA

Ejemplos:
— LIST 12ϕ
— LIST 3ϕ
— LIST

Cuando el argumento se omite, el intérprete BASIC ejecuta este comando a partir de la línea 1.

El listado del programa se visualiza en páginas de 22 líneas presentando en la parte inferior de la pantalla el mensaje:

scroll?

este mensaje, como ya recordará el lector de lo explicado anteriormente con la sentencia «PRINT», sirve para preguntarnos si queremos visualizar la siguiente página. Pulsando las teclas «N», «SPACE» o «STOP» («SYMBOL SHIFT» + «a») el listado se interrumpirá y se nos presentará el mensaje:

D BREAK — CONT repeats

pulsando cualquier otra tecla se visualizará la siguiente página y así sucesivamente hasta que se termine el listado (mensaje ϕ OK).

Cuando como argumento se introduce, por error, un número decimal, el intérprete BASIC redondea este valor hasta el número entero más próximo: si tiene un programa almacenado en la memoria, ejecute estos dos comandos directos y observe los resultados:

LIST 1ϕ.2
LIST 1ϕ.5

Si se especifica un número de línea inexistente, el comando «LIST» empezará a ejecutarse a partir de la siguiente.

El argumento también puede ser una variable numérica previamente definida. En el siguiente ejemplo, la instrucción «LIST línea» visualiza el listado a partir del valor asignado a la variable «línea».

```
1 REM *********
     * LISTAR *
     *********
2 INPUT "Numero de linea (10-50)";linea
3 LIST linea
10 REM "Linea 10"
20 REM "Linea 20"
30 REM "Linea 30"
40 REM "Linea 40"
50 REM "Linea 50"
```

LIST y EDIT

En el capítulo «2» dedicado a la edición de programas y corrección de los posibles errores, se estudió un método para corregir líneas de programa una vez editadas. Este método consistía en desplazar con los cursores (y) el prompt " > " hasta situarlo en la línea que queríamos corregir; pero ¿qué pasa si tenemos el cursor en la línea 4ϕϕϕ y queremos corregir la 2ϕ?, como vemos este método no es efectivo ya que perderíamos mucho tiempo desplazando cursores; en estos casos resulta más interesante utilizar el comando «LIST».

Para situar el prompt " > "en la línea que deseamos corregir basta simplemente con pedir un listado a partir de dicha línea, por ejemplo si deseamos corregir la línea 2ϕ, introduciremos el comando directo:

LIST 2ϕ

lógicamente si el listado es largo aparecerá el mensaje:

scroll?

pulsando la tecla «N», «SPACE» o «STOP» el listado se interrumpirá. El prompt ya lo tenemos situado en la línea 2ϕ; para corregirla utilizaremos la Función «EDIT», con lo que nos pasará a la parte inferior de la pantalla. A partir de este momento podremos corregirla utilizando los cursores (⇨y) y la función «DELETE». Una vez terminada la modificación, pulsando «ENTER» volverá a la parte superior. ■

SALTOS INCONDICONALES Y CONDICIONALES

Hay ocasiones en que por razones de estructura de un programa, interesa que las instrucciones no se ejecuten de forma secuencial, es decir, una detrás de otra, sino que, por el contrario realicen *saltos*; éstos pueden clasificarse dependiendo de su función en:

— INCONDICIONALES.
— CONDICIONALES.

Como su propio nombre indica, un salto *incondicional* es aquel que salta directamente al número de línea especificado en el argumento, sin embargo, los condicionales necesitan que se cumpla previamente la condición prevista en la instrucción.

Ejemplo sentencia "GO TO".

GO TO

Acceso al teclado




MODO K

Tipo de sentencia

Comando de programación.

Definición

La sentencia «GO TO» realiza los saltos incondicionales dentro de un programa, su estructura general es:

SENTENCIA	ARGUMENTO
GO TO	N.º de línea

Programa "Areas" menú de opciones.

PROGRAMA 1

```
10 REM **********
       * AGENDA *
       **********
20 PRINT AT 0,4;"NOMBRE    TELEFONO"
25 PRINT "_____       ________"
27 LET indice=1
30 INPUT "Nombre >>> "; LINE n$
40 IF n$="FIN" OR n$="fin" THEN GO TO 1000
50 PRINT n$,
60 INPUT "Telefono >>> ";telefono
70 PRINT telefono
80 IF indice=18 THEN GO TO 1000
90 LET indice=indice+1
100 GO TO 30
1000 PRINT #0;AT 1,2;"<<<<< Fin de edicion >>>>>"
1010 PAUSE 0
```


Programa "Areas" desarrollo opción "Rombo".

Los operadores *lógicos* «AND» y «OR» son utilizados cuando hay una combinación de condiciones dentro de una sentencia «IF ... THEN ...». El operador «AND» implica que deben cumplirse todas y cada una de las condiciones.

Ejemplo:

1ϕ IF n > = ϕ AND n < = 9 THEN ...

solamente se cumplen las dos condiciones, cuando la variable «n» tiene un valor comprendido entre «ϕ» y «9».

Utilizando el operador «OR» basta solamente con que se cumpla una de las condiciones previstas en la comparación:

Ejemplo:

1ϕ IF A$ = "A" OR B$ < > "C" OR n > 3ϕ THEN ...

en este ejemplo, se cumple la condición general en cualquiera de los siguientes casos:

a) Cuando la variable A$ tenga el valor "A".
b) Cuando el contenido de la variable B$ sea distinto de "C".
c) Cuando "n" sea superior a "3ϕ".
d) o con cualquier combinación de los anteriores casos.

También pueden combinarse los operadores «AND» y «OR».

Ejemplo:

1ϕ IF (J$ = "NO" AND n < 1ϕϕ) OR (J$ = "SI" AND (p = 1ϕ OR t = 7)) THEN ...

primero, se evalúan individualmente las condiciones encerradas entre paréntesis, y posteriormente, se evalúan entre sí los resultados parciales, por tanto, será necesario que se cumplan cualquiera de las siguientes condiciones:

a) Cuando J$ = "NO" y n < 1ϕϕ.
b) Cuando J$ = "SI" y p = 1ϕ.
c) Cuando J$ = "SI" y t = 7.
d) Combinaciones de a, b y c.

El programa número «1» es un ejemplo de aplicación de las sentencias «IF ... THEN ...»; acaba su ejecución cuando se teclean dieciocho nombres con sus correspondientes teléfonos o cuando se introduce la palabra «FIN» o «fin» en el instante que el ordenador espera un nombre.

Evaluación de las condiciones

Cuando una condición se cumple, es decir, que es verdadera, se le asigna el valor "1" (distinto de ϕ) y cuando es falsa, el valor ϕ (igual a ϕ). Para evaluar una condición compleja, primero se evalúan una a una, asignando los valores «ϕ» o «1» según corresponda; pos-

Ejemplos:

— GO TO 3Ø
— GO TO 7Ø

Una sentencia de este tipo transfiere la ejecución del programa a la línea especificada en su argumento.

En el siguiente programa, al analizar el intérprete BASIC la instrucción «4Ø», la siguiente que ejecutará será la «1Ø».

```
10 REM **************
       *            *
       *   GO TO    *
       *            *
       **************
20 INPUT "teclee lo que quiera
y pulse    ""ENTER""  >>>>> ";a$
30 PRINT a$
40 GO TO 10
```

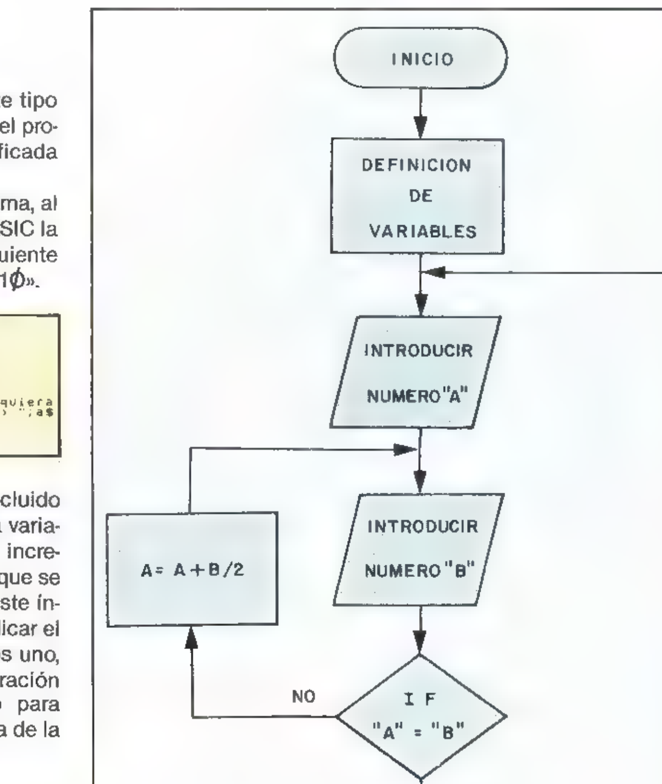
En este otro se ha incluido un *índice* asignado a la variable «potencia» que se incrementa en uno cada vez que se ejecuta la línea «5Ø», este índice sirve tanto para indicar el número de veces menos uno, que se realiza la operación matemática (2^n) como para utilizarse como potencia de la misma.

```
10 REM **************
       *            *
       * POTENCIAS  *
       *            *
       **************
20 LET potencia = 0
30 PRINT potencia;
40 PRINT ": ";2↑potencia
50 LET potencia=potencia+1
60 GO TO 30
```

Debe poner atención al calcular el número de línea donde desea que se realice el salto, ya que podrían no ejecutarse ciertas líneas intermedias.

Ejemplo:

```
10 REM ****************
       *              *
       * INCORRECTO   *
       *              *
       ****************
20 LET a$="MICRO"
30 LET b$="HOBBY"
40 LET c$="SEMANAL"
50 PRINT a$;
60 PRINT b$+" ";
70 PRINT c$''
80 GO TO 60
```



Ejemplo salto y ejecución condicional (IF... THEN...).

en este ejemplo, solamente la primera vez se ejecuta el programa correctamente; en las siguientes, la variable «a$» no se visualiza, por tanto la línea 8Ø debería ser:

GO TO 5Ø

«GO TO» también puede ser utilizado como comando directo, esta aplicación es bastante interesante, ya que permite ejecutar un programa sin alterar el contenido de las variables hasta ese momento definidas. En la depuración de programas se utiliza frecuentemente en sustitución del comando «RUN» que lo borra todo.

El argumento puede ser una variable de tipo numérico:

```
  1 REM *************
        *           *
        * VARIABLE  *
        *           *
        *************
  4 INPUT "A que linea hago el
""GO TO"" 10, 20,30,40,50 o 60?
";linea
  6 GO TO linea
 10 PRINT "salto a la 10"
 20 PRINT "salto a la 20"
 30 PRINT "salto a la 30"
 40 PRINT "salto a la 40"
 50 PRINT "salto a la 50"
 60 PRINT "salto a la 60"
 70 PRINT ''
 80 GO TO 4
```

una vez introducida la variable «línea» el control de ejecución salta hasta el valor especificado en ella.



IF ... THEN ...

Acceso al teclado

MODO K











Tipo de sentencia

Comando de programación.

Definición

El grupo de sentencias «IF» y «THEN» permiten realizar los saltos o ejecutar una serie de instrucciones de una manera *condicional*, es decir, en función del resultado de una comparación.

Las estructuras básicas son:

a) Salto condicional.

SENTENCIA	ARGUMENTO
IF condición	THEN GO TO...

Ejemplo:

```
10 REM **********
       *        *
       * SALTO  *
       *        *
       **********
20 INPUT "Numero ""A""? ";a
30 INPUT "Numero ""B""? ";b
40 IF a=b THEN GO TO 70
50 LET a=a+b/2
60 GO TO 30
70 PRINT "Valor ""A"" : ";a
80 PRINT '''"Fin"
```

cuando las variables «a» y «b» sean iguales, la ejecución del programa continuará en la línea 7Ø, si no, se asignará a la variable «a» el valor de «a + b/2» y posteriormente, ejecuta un salto incondicional a la línea 3Ø.

b) Ejecución condicional.

SENTENCIA	ARGUMENTO
IF condición	THEN instrucciones...

Ejemplo:

```
10 REM **************
       *            *
       * EJECUCION  *
       *            *
       **************
20 PRINT "MICROHOBBY"
30 INPUT "Quiere continuar(SI
/NO)? ";f$
40 IF f$="NO" THEN PRINT AT 10
,15;"FIN": LET a=0: LET b=0: STO
P
50 GO TO 20
```

si no se cumple que la variable de cadena «f$» sea igual a «NO» se ejecuta la instrucción siguiente; si por el contrario lo son, se ejecutan las sentencias que acompañan al «THEN».

De lo explicado hasta este momento, se desprende que la sentencia «IF ... THEN ...» es como una encrucijada con dos caminos, donde el ordenador tiene que elegir uno de ellos.

OBSERVACION

Si se utiliza la sentencia «GO TO», dentro de la lista de instrucciones que deben ejecutarse si se cumple la condición impuesta en el «IF», ésta deberá ser colocada la última, ya que de lo contrario, quedarían sin ejecutar algunas instrucciones.

Ejemplo:

```
1ØØ IF mes = 8 THEN PRINT
"AGOSTO" : GO TO 12Ø :
LET mes = Ø
```

la asignación del valor «Ø» a la variable «mes» no se realiza, ya que antes se ejecuta una instrucción de salto incondicional a la línea 12Ø.

Para realizar las comparaciones, puede utilizarse cualquiera de los operadores *relacionales*:

>	MAYOR QUE ...
<	MENOR QUE ...
> =	MAYOR O IGUAL
< =	MENOR O IGUAL
< >	DISTINTO

Ejemplos:

— IF a > b THEN ...
— IF J$ < S$ THEN ...
— IF n > = K * t THEN
— IF Kilo < = 7 THEN ...
— IF P$ < > "SI" THEN...



como el microdrive, como memoria externa. Además de esto, es importante señalar también que es compatible con los diferentes tipos de interface que se comercializan en la actualidad en el mercado español.

Se le ha dotado al programa, también, de dos nuevas opciones que le dan aún una mayor potencia: la primera, la de escribir textos personalizados y, la segunda, la posibilidad de imprimir repetidamente un texto con la particularidad de que cada uno puede colocarse en un campo diferente de todo un fichero escrito aparte.

El procesador de textos es el más completo que existe actualmente en el mercado y además, uno de los que se han hecho más versiones hasta la fecha.

El menú de opciones es muy completo, nos permite manipular un texto y hacer con él todo tipo de cosas: imprimir mayúsculas, mover el cursor en bloques, por palabras, por caracteres, por párrafos, scroll descendente y ascendente; centrado de líneas; inserción de líneas y caracteres; ensamblado de párrafos; justificación de márgenes; justificación de líneas; normalización de márgenes; impresión de textos; imprimir con el doble de alto; visión ampliada del texto; marcar el principio y el final del Block; Copy del Block; borrado del texto y modo de inserción.

Los que posean el interface 1, tienen una opción para hacer un catálogo del menú, ofreciéndonos una lista con los 50 primeros programas o archivos cargados en el Microdrive.

Otra opción, es la que nos permite elegir el Interface con el que vamos a trabajar: Centronics o RS232.

La V6 del Context permite trabajar con textos de hasta 300 líneas. Contiene una pantalla de información dentro del mismo programa que nos sirve de guía en todo momento, y nos sitúa en la posición exacta en la que está siendo impreso un carácter.

Puede también enviarse a la impresora cualquier código, bien sea subrayado, con el doble de alto o de cualquier otro tipo.

Admite además cualquier modelo de impresora con interface serie RS-232, Centronics, Seikonsa GP 50 y Zx-Printer.

Es, en definitiva, una buena versión del programa inglés que puede ser de mucha utilidad para todos aquellos que tienen que manejar textos muy a menudo, y quieran hacerlo de una forma cómoda, simple y práctica.

Curso didáctico para niños

TEORIA DE LA MUSICA I Y II

Music-Soft

48 K

Tipo: Didáctico Musical

PVP: 3.800 cada uno

Los programas musicales no abundan en nuestro país aunque son, sin embargo, una buena costumbre educacional además de una forma práctica de introducir al niño en el mundo de la música. Este curso para ordenador, está dividido en dos partes, cada una de las cuales es un programa diferente. El curso está pensado para que el alumno aprenda, de forma progresiva, a través de unas lecciones explicativas acompañadas de representación gráfica que le introducen, de forma grata, en el tema.



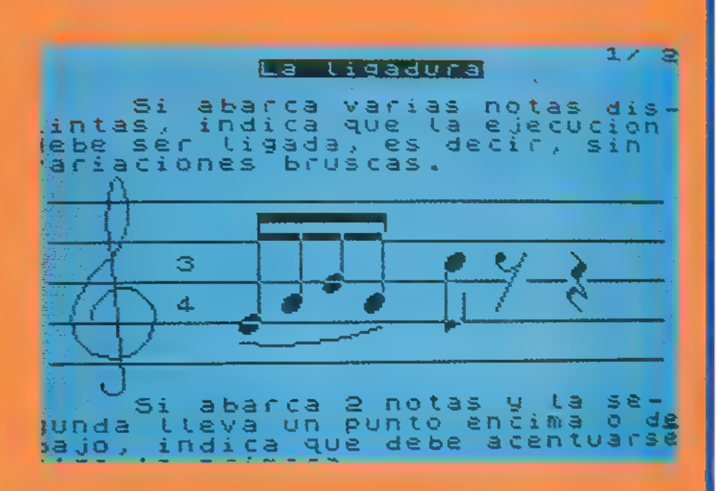
Cuando cargamos el programa, pasamos a la lección correspondiente donde se nos explica la materia concreta. Los cursos, que son muy completos, tratan cada uno de ellos los siguientes temas:

1º Curso: Introducción al lenguaje musical. Las claves. Los silencios. El compás. Las alteraciones. Repeticiones. Signos de prolongación. Los intervalos. Síncopas. Divisiones artificiales. La tonalidad. Dinámica.

2º Curso: Articulación. Notas de adorno. Escala, arpegios y acordes. Armadura de la tonalidad. Polirritmia. Instrumentos y afinaciones. El metrónomo. El sonido. Formación de ondas. El sintetizador. El secuenciador. El Sistema Midi.

Todos los temas están tratados con gran lujo de detalles y acompañados de buenas ilustraciones que dotan de un interés especial al curso y evitan que éste pudiera caer en la monotonía.

Los gráficos están bien y, didácticamente, ambos cursos están estructurados con mucha lógica y pensando siempre en el alumno.



En el programa han colaborado Paloma Lemanche, Luis Suja y Pierre Suja. Es un buen curso que puede servir para introducir en la música, de una forma diferente, a todos aquellos que hasta ahora creían que la música era sólo escuchar discos o ir a bailar a algún sitio.

INTERIORIDADES Y FUNCIONAMIENTO DE LA "ULA" (I)

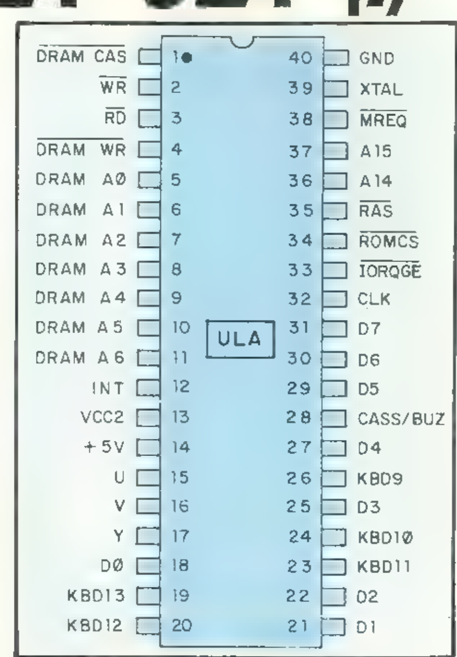
Primitivo de FRANCISCO

La ULA, ese extraño, negro y caluroso circuito integrado, tantas veces mencionado y nunca bien entendido, va a ser desmenuzado en nuestras páginas a fin de que Vd. le pierda rápidamente el respeto para pasar a ser un elemento de su sistema, tan útil y conocido como pueda ser el propio microprocesador.

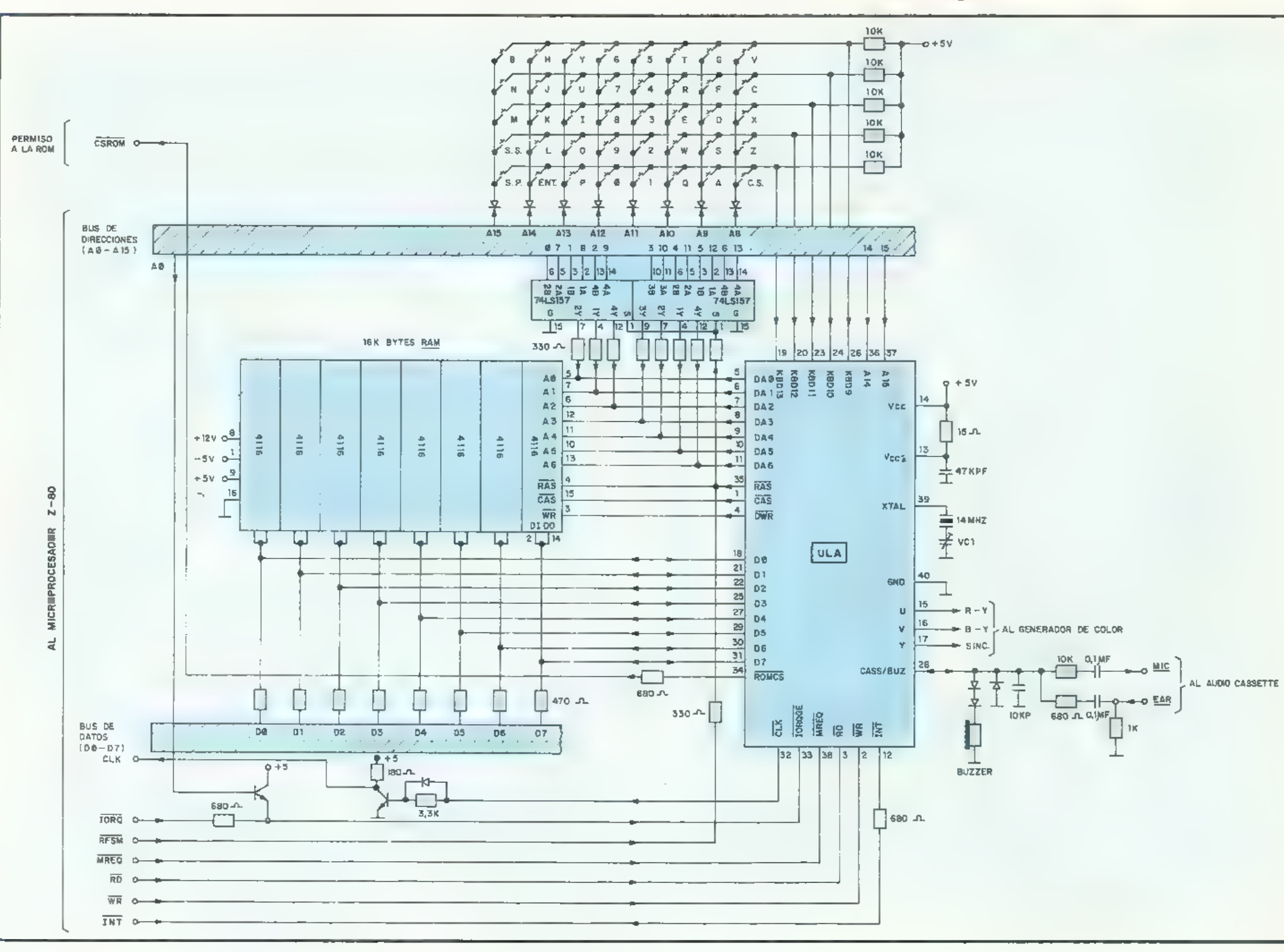
La ULA es un chip (término anglosajón equivalente a pastilla o circuito integrado monolítico) que ha sido diseñado por SINCLAIR RESEARCH para que el microprocesador se pueda relacionar con el cassette, teclado, altavoz o buzzer, monitor o salida de TV, RAM de pantalla y zona de variables del sistema.

La palabra ULA se corresponde con las siglas de UNCOMMITTED LOGIC ARRAY cuyo significado es: Matriz lógica no encargada. La ULA alberga en su interior, efectivamente, un conjunto de circuitos mayoritariamente lógicos, el chip es un «Custom» o circuito de encargo, diseñado por Sinclair, pero realizado por la firma Ferranti (la palabra uncommitted viene a indicar que si bien Sinclair no la realiza, sí posee todos los derechos).

La ULA es, en el ZX Spectrum, el chip que convierte a este micro en específico y, por tanto, incopiable si no se dispone de la ULA correspondiente, la cual, evidentemente, no está comercializada ya que, como se ha dicho, es un producto



Distribución de señales y alimentación de la ULA. (Fig. 1).



Conexión de la ULA a la RAM de 16KBytes y al resto del sitema. (Fig. 2).

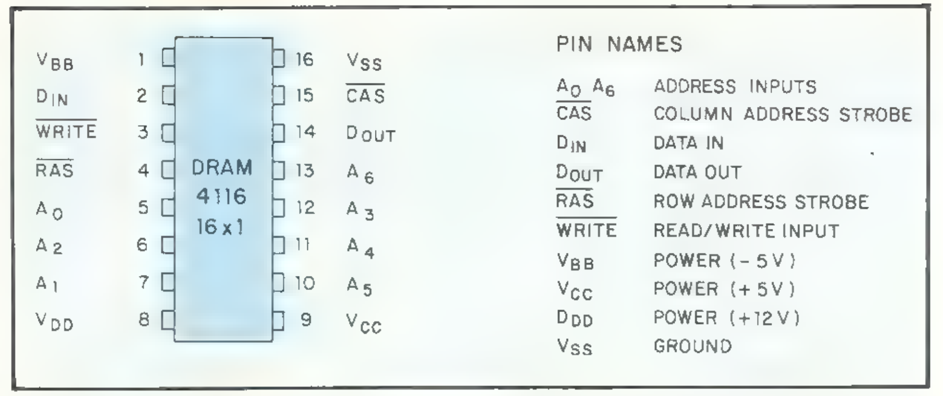
propio de Sinclair, que la incluye en sus Spectrum, unidad a unidad.

El uso de la ULA en el Spectrum hace que la circuitería lógica para el acceso a teclado, cassette y pantalla, se encuentre disminuida al máximo al estar toda ella concentrada en este chip multifuncional.

Analizándola más profundamente, se observa que ha sido diseñada conceptualmente con bastante eficacia, aunque para su total desarrollo han sido precisas varias ediciones, lo cual es fácil de observar leyendo los diferentes códigos numéricos de las ULA de los Spectrums que hemos abierto. Algo que hasta la fecha no ha quedado suficientemente resuelto, es el desmesurado calor que ha de disipar el CHIP, consecuencia directa del gran volumen de circuitos que ha sido preciso integrar para que la ULA cumpla la totalidad de cometidos mencionados.

Aprovechando este comentario en torno a la temperatura, aconsejamos, en base a la experiencia, que se pegue algún disipador a la pastilla de la ULA o simplemente un trozo de aluminio del tamaño de la misma con el fin de ayudarle en la evacuación del calor. En algunos casos de fallos espontáneos de algunos Spectrums, hemos comprobado que ésta era la solución; de todas formas, ayudarle a disipar siempre es bueno y puede colaborar a aumentar su vida.

En resumen, la ULA es un LSI (Larga escala de integración) que auxilia al Z-80 proporcionándole el adecuado acceso a los principales periféricos, es, por tanto, el segundo de a bordo en el Spectrum, empleando jerga marinera.



Distribución de señales a los terminales de la DRAM (RAM dinámica) 4116 (16 KBits). (Fig. 3).

Bloques funcionales de la ULA

Podemos dividir la funcionalidad de la ULA en cinco bloques, cada uno de los cuales gestiona el acceso a los diferentes periféricos.

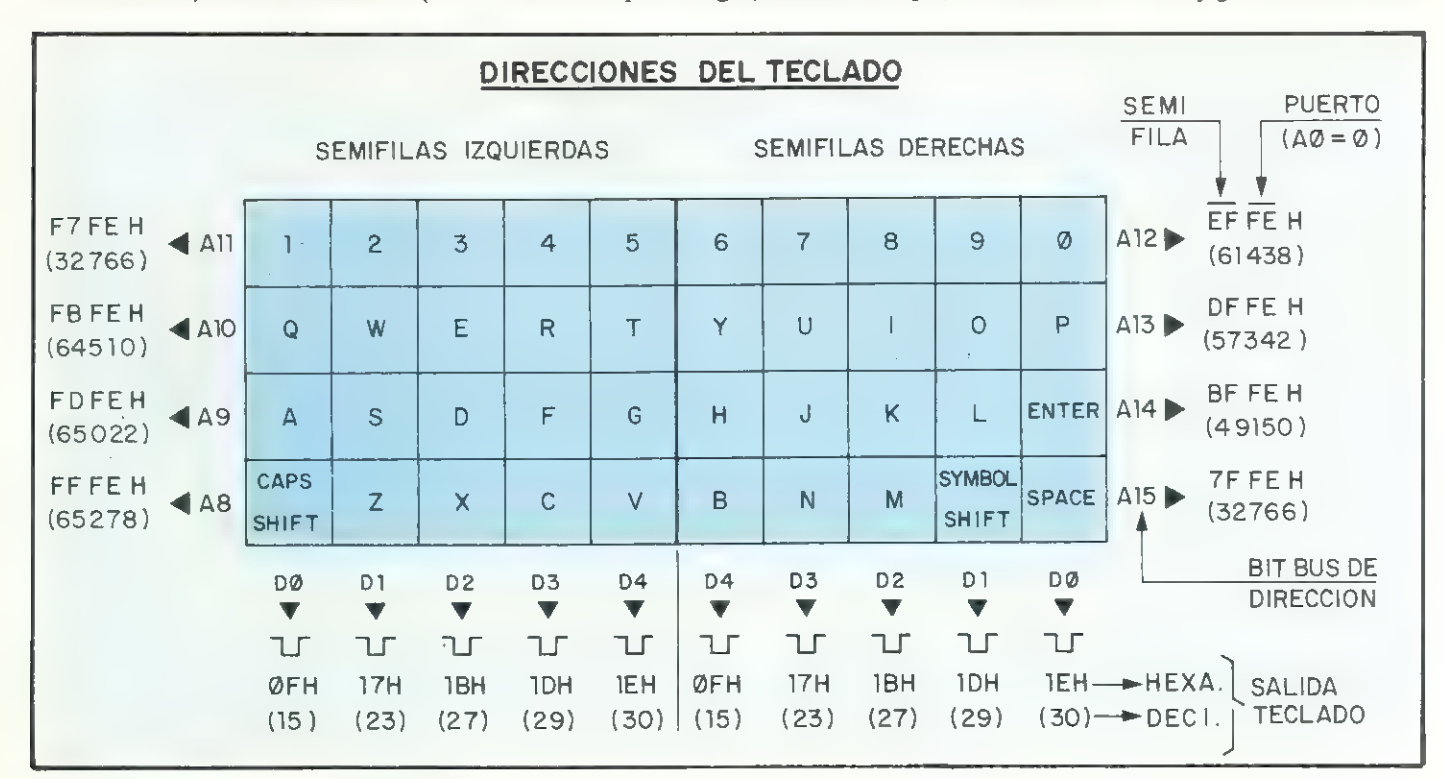
Aclararemos que estamos empleando el término de periférico a todo dispositivo externo al microprocesador que tiene una funcionalidad específica. Excluimos, por tanto, al bloque de memoria sin la cual no existiría un soporte físico para el software que ha de ejecutar el microprocesador.

El primer bloque de la ULA, es el encargado de posibilitar la exploración del teclado; el segundo bloque, es el que permite el acceso a cassette, tanto para salvar como para cargar; el tercer bloque, se ocupa de generar el sonido que se reproduce por el minialtavoz interno; el cuarto bloque, cumple la función de DMA o acceso directo a memoria (es usado, por ejemplo, para leer constantemente el fichero de presentación por pantalla, actualización de ciertas variables, etc.), y el quinto bloque, que se ocupa de generar las señales eléctricas de vídeo (luminancia, color y sincronismos).

Los cinco bloques, aunque independientes a nivel funcional, están interrelaciones por las señales de control del microprocesador mediante la lógica interna.

En la *figura número dos,* se muestra mediante el esquema eléctrico, la totalidad de conexiones de la ULA a sus circuitos y periféricos dependientes.

La ULA se encuentra encapsulada en plástico en formato DIL (Dual in line) de 40 patitas o pines. La distribución de señales se ve en la *figura número uno.*



El teclado: Direcciones de sus semifilas y salidas de sus columnas hacia el bus de datos. (Fig. 4).

Conexión eléctrica de la ULA

Volvamos a la figura número dos, en ella se muestran las conexiones de la ULA al resto del microordenador.

En la parte superior, existen cinco terminales de entrada denominados KBD 9 a KBD13 que proceden directamente del teclado, por ellos, entrará la información de la tecla pulsada cuando lo requiera el programa. Por los terminales denominados A14 y A15 del bus de direcciones, la ULA se entera si el microprocesador está corriendo en los primeros 16 KBytes de RAM con los cuales opera en DMA (de esto vamos a tratar en un próximo artículo).

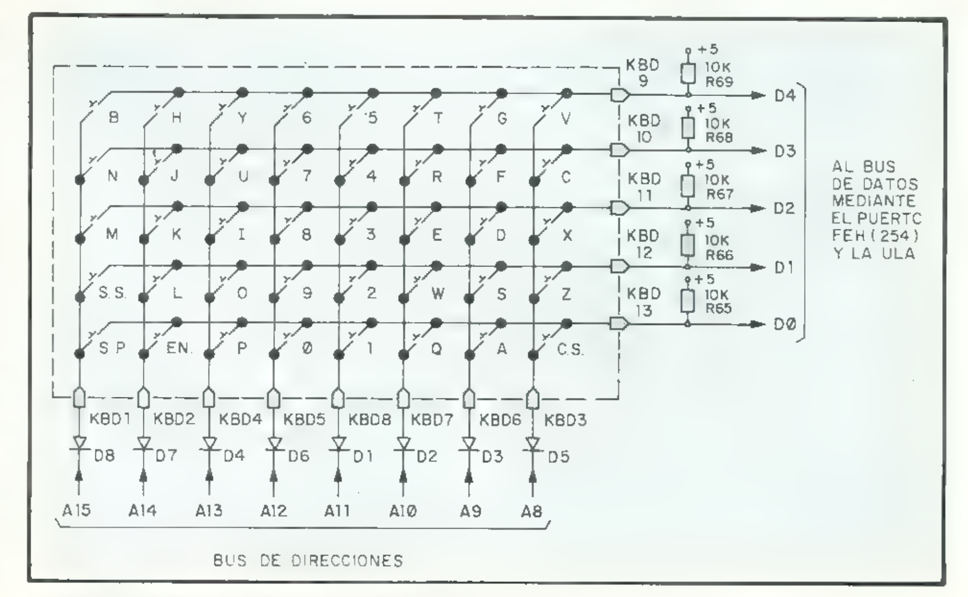
Las entradas denominadas Vcc, Vcc2 y GND, son entradas de alimentación. XTAL es el terminal que va asociado al cristal de cuarzo de 14 Mhz, verdadero corazón del Spectrum, bajo cuyo ritmo operan el microprocesador y la ULA. Esta entrada es, en realidad, el control del oscilador interno que proporciona el clock del sistema. Los terminales U y V entregan las señales de información de color para cada punto de pantalla barrido por el televisor o monitor de vídeo, y es la salida de luminancia ỹ sincronimos de la señal de vídeo.

Por el terminal CASS/BUZ, circulan las señales de cassette o la señal sonora hacia el minialtavoz o Buzzer. Las señales MREQ, RDWR e INT proceden del bus de control del microprocesador. IORQGE es el terminal por el cual se direcciona la ULA. EL Z-80 accede a la ULA mediante el bit AØ del bus de direcciones y la señal IORQ cuando ambos son cero, la ULA se direcciona como puerto. Por el terminal CLK, la ULA inyecta la señal de reloj al Z-80, previamente dividida en frecuencia. El hecho de que el clock del microprocesador atraviese la ULA, determina que ésta pueda controlarlo, como también veremos detenidamente.

El terminal ROMCS bloque la ROM cuando es preciso dejar su bus en alta impedancia. Los terminales DØ a D7 van al bus de datos. Mediante los terminales DAØ a DA6 y RAS, CAS y DWR, la ULA obtiene el acceso directo a memoria al tiempo que colabora en el refresco de los primeros 16K Bytes de memoria RAM dinámica.

El teclado

El teclado es el primer periférico controlado por la ULA que vamos a analizar. El teclado es eléctricamente una matriz de pulsadores de ocho columnas por cinco filas, lo cual corresponde a cuarenta teclas. Estas teclas así organizadas, son válidas tanto para el Spectrum normal como para el Spectrum Plus. En el Spectrum Plus el resto de las teclas son, en realidad, contactos en paralelo con las cuarenta básicas que desdoblan la operatividad de algunas de ellas y, por tanto, facilitan su manejo. Por ejemplo, la función DELETE, que en el Spectrum normal requiere el uso de dos teclas simultáneamente (CAP SHIFT y Ø), en el PLUS, la tecla independiente DELETE activa los dos contactos internos necesarios al ser oprimida.



Esquema eléctrico del teclado del Spectrum. (Fig. 5).

En ambas versiones del Spectrum, los pulsadores de las teclas están realizados por dos láminas de plástico con pistas metalizadas que se tocan cuando la tecla respectiva es actuada.

Las columnas del teclado van conectadas eléctricamente a los bits A8 a A15 del bus de direcciones, tal como se muestra en la *figura número cinco.* Cuando una tecla es oprimida, la señal del bit del bus de direcciones correspondiente entra en el bus de datos controlada previamente por la ULA.

Los diodos puestos en serie con cada uno de los bit de dirección, hacen que no se produzcan cortocircuitos entre ellos cuando dos o más teclas son activadas a la vez. Los terminales del teclado se conectan directamente a la tarjeta del Spectrum mediante dos conectores, por uno las filas y por el otro las columnas. Estos terminales están numerados desde KBD 1 hasta HBD13.

En la salida del teclado, se hallan cinco resistencias de 10K Ohmios que polarizan positivamente las salidas del teclado cuando no se ha oprimido ninguna tecla; por el contrario, cuando se activa alguna de ellas, el cero que genera el bus de dirección atraviesa el diodo y tras la tecla se presenta en la salida correspondiente poniéndola a cero.

Pongamos un ejemplo: Supongamos que se oprime la tecla R; cuando la rutina de exploración del teclado presente un cero en el bit A10 del bus de direcciones y el contacto de la tecla esté cerrado, el bit D3 será puesto a cero. Este cero llegará, finalmente, al bus de datos mediando la ULA que quedará a su vez habilitada por la propia rutina de exploración mediante el bit AØ = Ø.

Direccionamiento del teclado

El teclado puede direccionarse mediante la rutina de exploración contenida en la ROM, o bien, directamente usando el puerto FE H (254) e interpretando después el dato recibido.

El teclado se encuentra dividido en dos bloques de semifilas. Cada una de las semifilas tiene cinco teclas y se corresponde con una dirección de puerto. Así, para todo el teclado se utilizan ocho semifilas.

En ambas semifilas, el dato se recoge con los cinco bits de menor peso del bus de datos.

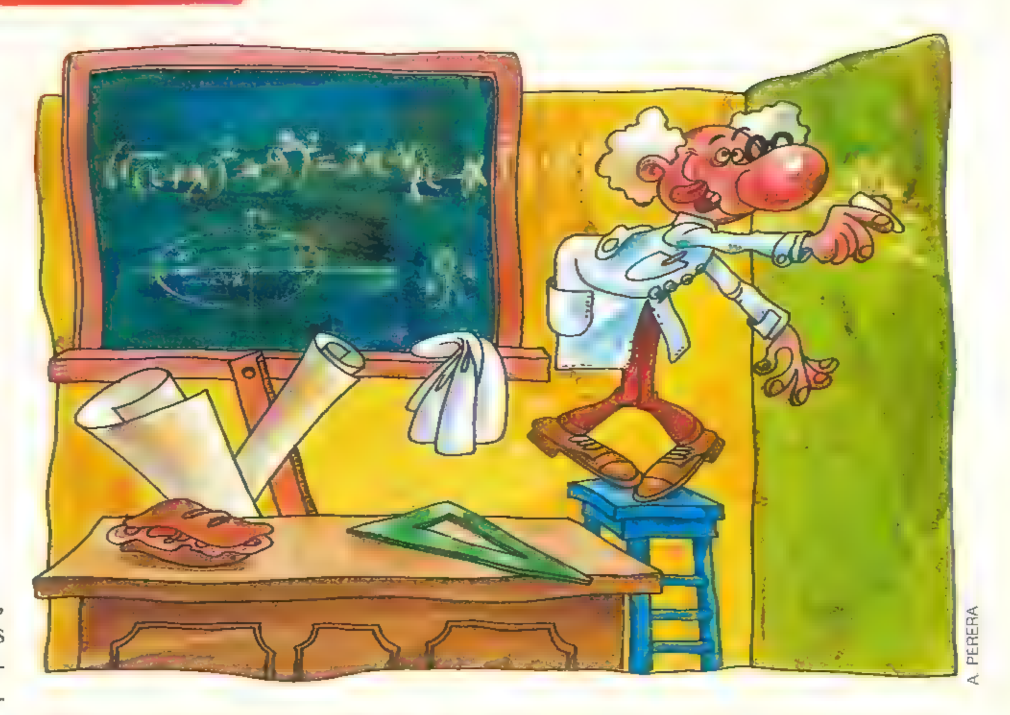
La dirección de semifila se compone de dos bytes, el de menor peso ha de ser siempre FEH (en FEH el Bit AØ = Ø) y el byte de mayor peso es el que activa la semifila correspondiente. En la *figura número cuatro* se muestran la totalidad de las direcciones y los datos que se obtienen, direcciones y datos están expresados en decimal y hexadecimal para comodidad del usuario.

HELP

Jesús P. BENITEZ VERAUGO

Spectrum 48 K

Este programa de utilidades, le servirá de ayuda para realizar cualquier tipo de cálculo, reuniendo, en una misma cinta, una gran variedad de opciones aritméticas.



A. PERERA

Determinantes, sistemas de ecuaciones, ecuaciones de segundo grado, ecuaciones en general, gráficas de funciones, integrales definidas, operaciones normales, productos vectoriales, escalares y módulos de vectores. Todas estas opciones se pueden realizar de la manera más sencilla, una vez que se presenta en pantalla el menú de opciones principal. Con sólo introducir nuestra elección, el ordenador nos pedirá los datos oportunos y nos dará los resultados. Para los gráficos, ofrece dos alternativas:

"1-AMPL", pulsando 1 ampliará las gráficas de dos formas.

"2-NEW LIMITS", que cambiará los límites de la función.

Premiado con 15.000 pts.



```
  10 REM      © J. BENITEZ 1984
  20 BORDER 0: PAPER 0: INK 7
  35 CLS
  40 PRINT AT 0,2;"1- DETERMINAN
TES"
  50 PRINT AT 2,2;"2- SISTEMAS D
E ECUACIONES"
  60 PRINT AT 4,2;"3- ECUACIONES
 DE GRADO 2"
  70 PRINT AT 6,2;"4- ECUACIONES
 EN GENERAL"
  80 PRINT AT 8,2;"5- GRAFICAS"
  90 PRINT AT 10,2;"6- INTEGRALE
S DEFINIDAS"
 100 PRINT AT 12,2;"7- OPERAR"
 110 PRINT AT 14,2;"8- PRODUCTO
VECTORIAL"
 112 PRINT AT 16,2;"9- PRODUCTO
ESCALAR"
 115 PRINT AT 18,2;"10- MODULO D
E VECTORES"
 117 PRINT AT 20,2;"11- SALIR"
 120 PRINT AT 21,4; FLASH 1;"INT
RODUZCA OPCION"
 130 INPUT op
 135 CLS
 140 IF op=11 THEN STOP
 141 IF op=1 THEN GO SUB 170
 142 IF op=8 THEN GO SUB 710
 143 IF op=3 THEN GO SUB 1000
 144 IF op=4 THEN GO SUB 590
 145 IF op=7 THEN GO SUB 830
 146 IF op=2 THEN GO SUB 3000
 147 IF op=10 THEN GO SUB 2000
 148 IF op=9 THEN GO SUB 2080
 149 IF op=5 THEN GO SUB 2500
 150 IF op=6 THEN GO SUB 1180
 160 GO TO 35
 165 REM         DETERMINANTES
 170 PRINT "            DETERMINANT
ES"
 175 LET ind=0
 180 INPUT "ORDEN DEL DETERMINAN
TE 2,3,4  ";d
 182 PRINT AT 4,0;"
                     "
 185 IF d<2 OR d>4 THEN PRINT AT
 4,0; FLASH 1;"NO ESTOY PROGRAMA
DO PARA HACERLO": GO TO 180
 190 DIM a(d,d)
 200 FOR i=1 TO d
 210 FOR j=1 TO d
 220 INPUT "A(";(i);",";(j);") "
;a(i,j)
 230 NEXT j
 240 NEXT i
 250 IF d=2 THEN GO TO 280
 260 IF d=3 THEN GO TO 350
 270 GO TO 431
 280 PRINT AT 6,6;"| ";a(1,1);"
";a(1,2);" |"
 290 PRINT AT 7,6;"| ";a(2,1);"
";a(2,2);" |"
 295 LET B=a(1,1)*a(2,2)-a(1,2)*
a(2,1)
 296 IF ind=1 THEN RETURN
 300 PRINT AT 12,6;"DETERMINANTE
= ";b
 310 PRINT AT 16,4;"OTRO DETERMI
NANTE ? (s/n)"
 320 PAUSE 0
 330 IF INKEY$="s" OR INKEY$="S"
 THEN CLS : GO TO 170
 340 RETURN
 350 PRINT AT 6,6;"| ";a(1,1);"
";a(1,2);" ";a(1,3);" |"
 360 PRINT AT 7,6;"| ";a(2,1);"
";a(2,2);" ";a(2,3);" |"
 370 PRINT AT 8,6;"| ";a(3,1);"
";a(3,2);" ";a(3,3);" |"
 380 LET B=(a(1,1)*a(2,2)*a(3,3)
+a(1,2)*a(2,3)*a(3,1)+a(2,1)*a(3
,2)*a(1,3))-(a(1,3)*a(2,2)*a(3,1
)+a(1,1)*a(2,3)*a(3,2)+a(3,3)*a(
1,2)*a(2,1))
 385 IF ind=1 THEN RETURN
 390 PRINT AT 12,4;"DETERMINANTE
= ";B
 400 PRINT AT 16,4;"OTRO DETERMI
NANTE ? (s/n)"
 410 PAUSE 0
 420 IF INKEY$="s" OR INKEY$="S"
 THEN CLS : GO TO 170
 430 RETURN
 431 PRINT AT 6,6;"| ";a(1,1);"
";a(1,2);" ";a(1,3);" ";a(1,4);"
 |"
 432 PRINT AT 7,6;"| ";a(2,1);"
";a(2,2);" ";a(2,3);" ";a(2,4);"
 |"
 433 PRINT AT 8,6;"| ";a(3,1);"
";a(3,2);" ";a(3,3);" ";a(3,4);"
 |"
 434 PRINT AT 9,6;"| ";a(4,1);"
";a(4,2);" ";a(4,3);" ";a(4,4);"
 |"
 435 LET b=0
 440 LET o=1: LET p=2: LET q=3:
LET r=4: GO SUB 520
 450 LET o=3: LET p=1: LET q=2:
LET r=4: GO SUB 520
 460 LET o=1: LET p=4: LET q=2:
LET r=3: GO SUB 520
 470 LET o=2: LET p=3: LET q=1:
LET r=4: GO SUB 520
 480 LET o=4: LET p=2: LET q=1:
LET r=3: GO SUB 520
 490 LET o=3: LET p=4: LET q=1:
LET r=2: GO SUB 520
 495 IF ind=1 THEN RETURN
 500 PRINT AT 12,4;"DETERMINANTE
= ";b
 510 GO TO 550
 520 LET a=a(1,o)*a(2,p)-a(1,p)*
a(2,o)
 530 LET b=a*(a(3,q)*a(4,r)-a(3,
r)*a(4,q))+b
 540 RETURN
 550 PRINT AT 16,4;"OTRO DETERMI
NANTE ? (s/n)"
 560 PAUSE 0
 570 IF INKEY$="s" OR INKEY$="S"
 THEN CLS : GO TO 170
 580 RETURN
 585 REM         ECUACIONES  N
 590 PRINT AT 2,2;"    ECUACIONE
S ORDEN N"
 600 INPUT "ECUACION (X)= ";e$
 610 INPUT "INTERVALO ";a;" ";b
 620 INPUT " STEP =";e
 630 FOR x=a TO b STEP e
 635 POKE 23692,3
 640 LET s=VAL e$
 645 PRINT s,x
 650 IF s<.001 AND s>-.001 THEN
PRINT FLASH 1;"solucion  pres co
nt...": PAUSE 0
 660 NEXT x
 670 PRINT "OTRA ECUACION ? (s/n
)"
 680 PAUSE 0
 690 IF INKEY$="s" OR INKEY$="S"
 THEN CLS : GO TO 590
 700 CLS : RETURN
 705 REM   PRODUCTO VECTORIAL
 710 PRINT AT 2,2;"    PRODUCTO
VECTORIAL"
 720 PRINT AT 6,0;"(ai + bj +ck)
x(di + ej + fk)"
 730 INPUT "a=";a,"b=";b,"c=";c,
"d=";d,"e=";e,"f=";f
 740 PRINT AT 6,0;"(";a;"i + ";b
;"j + ";c;"k)x(";d;"i + ";e;"j +
 ";f;"k)="
 750 LET i=b*f-c*e
 760 LET j=c*d-a*f
 770 LET k=a*e-b*d
 780 PRINT AT 7,4;"= ";i;"i + ";
j;"j + ";k;"k"
 790 PRINT AT 10,4;"otro product
o ? (s/n)"
 800 PAUSE 0
 810 IF INKEY$="s" OR INKEY$="S"
 THEN CLS : GO TO 710
 820 CLS : RETURN
 825 REM         OPERAR
 830 PRINT " STOP  para parar"
 835 INPUT LINE o$
 840 IF o$=" STOP " THEN RETURN
 850 PRINT VAL o$,
 860 GO TO 835
1000 REM          ECUACIONES  2
1020 CLS : PRINT AT 7,10;"ax↑2+b
x+c=0"
1030 PRINT AT 4,7;"teclee coefic
ientes"
1040 INPUT "a=";a,"b=";b,"c=";c
1050 LET d=(ABS b)↑2-4*a*c
1060 PRINT AT 12,4;a;"x↑2+(";b;"
)x+(";c;")"
1070 IF d<0 THEN LET d=ABS d: GO
 TO 1140
1080 PRINT AT 17,4;"x=";(-b+(SQR
 d))/(2*a)
1090 PRINT AT 19,4;"x=";(-b-(SQR
 d))/(2*a)
1100 PRINT AT 21,4;"otra ecuacio
n ?"
```



```
1105 PAUSE 0
1110 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 1000
1130 RETURN
1140 PRINT AT 14,4;"SOLUCIONES I
MAGINARIAS"
1150 PRINT AT 16,4;"X=";-b/(2*a)
;"+";(SQR d)/(2*a);"i"
1160 PRINT AT 18,4;"X=";-b/(2*a)
;"-";(SQR d)/(2*a);"i"
1170 GO TO 1100
1175 REM         INTEGRAR
1240 INPUT "funcion (x) ";f$
1250 INPUT "limites=? (2) ";a;"
 ";b
1260 INPUT "particion ";c: LET c
=2*c
1270 LET x=a: GO SUB 1350: LET i
=y
1280 LET d=(b-a)/c: LET o=c/2
1290 LET x=x+d: GO SUB 1350: LET
 i=i+y*4
1300 LET x=x+d: GO SUB 1350: LET
 i=i+y*2
1310 LET o=o-1: IF o<>0 THEN GO
TO 1290
1320 LET x=b: GO SUB 1350: LET i
=i-y
1330 CLS : PRINT AT 10,4;f$;AT 1
4,4;"limites= ";a;" , ";b;AT 17,
4;"I= ";d*i/3
1340 GO TO 1370
1350 LET y=VAL f$
1360 RETURN
1370 PRINT AT 19,4;"OTRA INTEGRA
L ?": PAUSE 0: IF INKEY$="s" OR
INKEY$="S" THEN GO TO 1180
1380 RETURN
1900 REM   MODULO  DE VECTORES
2000 CLS : PRINT "   MODULO DE V
ECTORES"
2010 PRINT AT 10,4;"v= ai +  bj
+ ck"
2020 INPUT "a= ";a,"b= ";b,"c= "
;c
2030 PRINT AT 12,4;"v= ";a;"i +
";b;"j + ";c;"k"
2040 LET mdl=SQR (a*a+b*b+c*c)
2050 PRINT AT 16,6;"[v]= ";mdl
2060 PRINT AT 20,5;"otro modulo
? (s/n)": PAUSE 0: IF INKEY$="s"
 OR INKEY$="S" THEN GO TO 2000
2070 RETURN
2075 REM    PRODUCTO ESCALAR
2080 PRINT AT 2,2;"    PRODUCTO
ESCALAR"
2090 PRINT AT 6,0;"(ai + bj +ck)
x(di + ej + fk)"
2100 INPUT "a=";a,"b=";b,"c=";c,
"d=";d,"e=";e,"f=";f
2110 PRINT AT 6,0;"(";a;"i + ";b
;"j + ";c;"k)*(";d;"i + ";e;"j +
 ";f;"k)="
2120 LET pe=a*d+b*e+c*f
2130 PRINT AT 7,4;"= ";pe
2140 PRINT AT 8,8;"otro producto
 ? (s/n)"
2150 PAUSE 0
2160 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 2080
2170 RETURN
2500 REM            GRAFICAS
2505 CLS
2510 INPUT "X(t) ";X$
2520 INPUT "Y(t) ";Y$
2525 LET f=1: LET rd=1
2530 INPUT "INTERVALO DE T (-t,t
) ";a
2540 PLOT 127,0: DRAW 0,175
2550 PLOT 0,87: DRAW 255,0
2570 FOR t=-a/rd TO a/rd STEP .1
/f
2580 LET ym=(VAL y$*f)+87
2590 LET xm=(VAL x$*f)+127
2600 IF xm<0 THEN GO TO 2640
2610 IF xm>255 THEN GO TO 2640
2620 IF ym<0 THEN GO TO 2640
2630 IF ym>127 THEN GO TO 2640
2635 PLOT xm,ym
2640 NEXT t
2650 PRINT #0;"PRESS   1-AMPL.
2-NEW LIMITS"
2660 PAUSE 0
2670 IF INKEY$="1" THEN INPUT "F
ACTOR ";f: PRINT #0;"       REDUZ
CO LIMITES ?": GO SUB 2800: CLS
: GO TO 2540
2680 IF INKEY$="2" THEN CLS : GO
 TO 2530
2685 PRINT #0;"             OTRA  C
URVA ?        "
2690 PAUSE 0
2700 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 2500
2710 CLS : RETURN
2800 PAUSE 0: IF INKEY$="s" OR I
NKEY$="S" THEN LET rd=f
2810 RETURN
3000 REM SISTEMAS DE ECUACIONES
3005 CLS
3010 INPUT "N DE INCOGNITAS (2,3
,4) ";N: DIM C(N,N): DIM I(N): D
IM A(N,N): DIM R(N)
3020 LET N=INT N: IF N>4 THEN GO
 TO 3010
3030 IF N<2 THEN GO TO 3010
3035 PRINT AT 15,0;"INTRODUZCA C
OEFICIENTE DE LA            INCOG
NITA"
3040 FOR J=1 TO N
3050 PRINT AT 16,18;J
3060 FOR I=1 TO N
3070 INPUT "DE ECUACION ";(I);"
";C(I,J)
3080 NEXT I
3090 NEXT J
3100 CLS
3110 PRINT AT 15,0;"INTRODUZCA T
ERMINO INDEPENDIENTE"
3120 FOR I=1 TO N
3130 INPUT "DE ECUACION ";(I);"
";I(I)
3140 NEXT I
3150 CLS
3152 FOR J=1 TO N: FOR I=1 TO N
3153 LET A(I,J)=C(I,J)
3155 NEXT I: NEXT J
3160 LET IND=1: IF N=2 THEN LET
DI=295
3165 IF N=3 THEN LET DI=380
3168 IF N=4 THEN LET DI=435
3170 GO SUB DI
3180 LET CO=B: IF CO=0 THEN PRIN
T AT 10,0; INVERSE 1;"  NO FORMA
 SISTEMA LA MATRIZ DE   COEFICIE
NTES VALE 0              ": GO TO 3
350
3190 FOR V=1 TO N
3200 FOR J=1 TO N
3210 FOR I=1 TO N
3220 LET A(I,J)=C(I,J)
3230 NEXT I
3240 NEXT J
3250 FOR J=1 TO N
3260 LET A(J,V)=I(J)
3270 NEXT J
3280 GO SUB DI
3290 LET R(V)=B
3300 NEXT V
3310 FOR J=1 TO N
3320 LET R(J)=R(J)/CO
3330 PRINT AT 10+(2*J),2;"SOLUCI
ON INCOGNITA ";J;" = ";R(J)
3340 NEXT J
3350 PRINT #0;"      OTRO SISTEMA
 ?"
3360 PAUSE 0
3370 IF INKEY$="S" OR INKEY$="s"
 THEN GO TO 3000
3380 RETURN
```

EL BOMBARDERO

Premiado con 15.000 pts.

José M.ª SILVA MURILLO

Spectrum 16 K

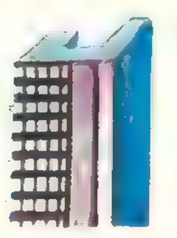


Una vez más, nos encontramos a bordo de un avión en que nosotros somos el piloto y tenemos que intentar superar una difícil situación que puede acabar trágicamente.

Sobrevolamos una gran ciudad de altos edificios que deberemos destruir sin remedio si queremos concluir nuestro cometido y lograr aterrizar. La emergencia es grande y nuestra misión también.

Para lograrlo, debemos disparar nuestras bombas disponiendo de tres disparos en cada pasada que demos sobrevolando los rascacielos. El mando para efectuar dicha operación es «Ø». Mucha suerte.





Pablo

LAS TORRES

Alejandro LABORJA y Jorge LIZAR

Spectrum 48 K

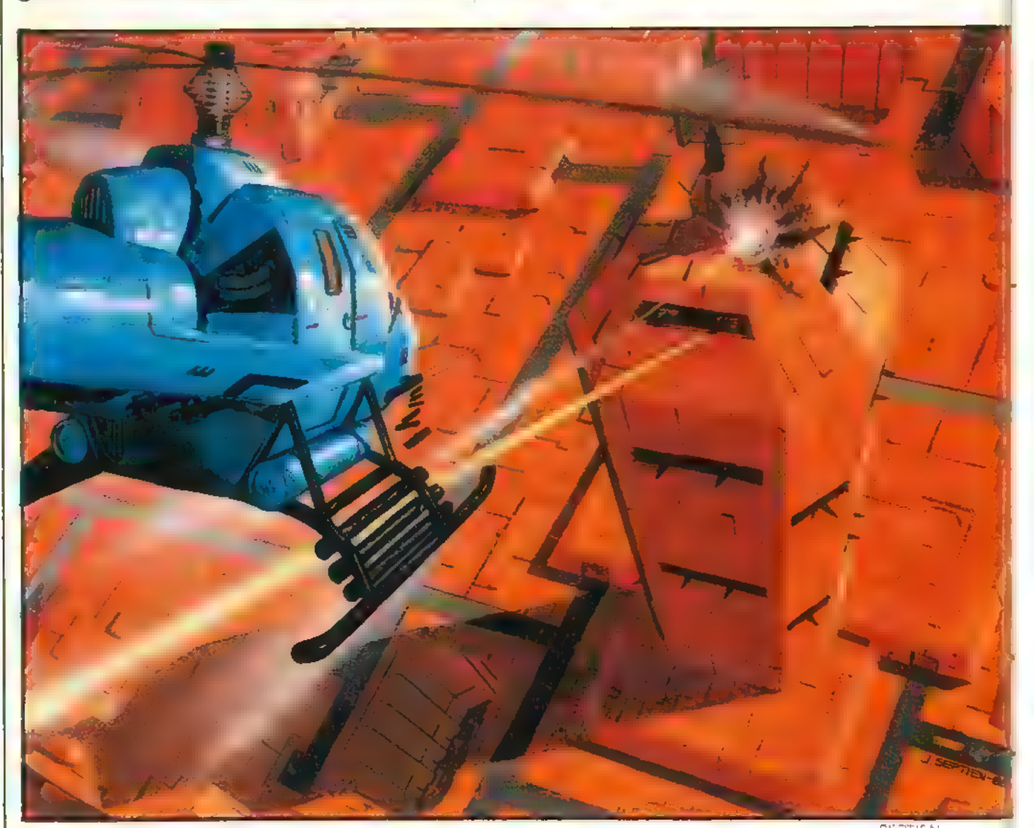
Somos, ahora, los artífices de una importante contraofensiva en una zona militar de alto secreto, y nuestra misión consiste en defender la zona del ataque enemigo que ha ocupado unas grandes torres desde donde nos atacan incansablemente.

Nosotros conducimos un helicóptero desde el que debemos disparar, e ir eliminando a los adversarios de las doce torres ocupadas.

Pulsando GOSUB, quedarán definidos los gráficos necesarios para el programa. Antes de introducir éste en el ordenador, si se hace por teclado, hay que meter las líneas 9ØØØ a 911Ø y hacer RUN.

Mediante la tecla «L» accedemos al disparo y, conforme vayamos eliminando invasores, irán sucediéndose las torres en la pantalla, hasta acabar con todos ellos. A continuación, en la línea 7ØØØ, encontramos la subrutina en la que se cambia la mejor puntuación, y se reinicia el programa.

Mediante la variable FUEL, se nos indica el combustible que tiene la nave; con NAVES, el número de vidas que le quedan al jugador; con AF, se define la puntuación obtenida hasta el momento, y con PANT, el número de línea (sumándolo a 3ØØØ tras haberlo multiplicado por 1ØØ) a la que se ha de acudir para leer los datos de la correspondiente pantalla.



Premiado con 15.000 pts.

Para aquellos a los que el juego les resulta demasiado fácil, basta con añadir la línea 1585 IF F(X)=PFN+1 THEN GOTO 65ØØ.

NOTAS GRAFICAS

```
L O S T U
```

```
    5 POKE 23624,50
   10 PRINT ;"TIENES QUE CONSEGUIR ATERRIZAR, PARA ELLO TIENES QUE DESTRUIR  LOS EDIFICIOS ";"PULSANDO ""0""","SOLO DISPONES DE 3 DISPAROS POR PASADA.";"SUERTE"
   80 LET pp=0
   90 GO SUB 3000
  100 GO SUB 300
  110 GO TO 2000
  200 LET x=20-a: REM disparo
  210 FOR t=0 TO x
  220 PRINT PAPER 5;AT a,b;" ": PRINT PAPER 5;AT a+t,b;" ": PAUSE 2: BEEP 0.03,0.0002: PRINT AT a+t,b;" ": PRINT AT a,b;" ": PRINT INK 4;AT 20,b;"█"
  225 IF ATTR (a+t+1,b)=7 THEN LET pp=pp+1: LET pt=pp+pp: PRINT " Puntos ";pt
  230 NEXT t
  240 RETURN
  300 REM graficos avion
  310 POKE USR "s",  BIN 00000111
  320 POKE USR "s"+1,BIN 01000011
  330 POKE USR "s"+2,BIN 01000001
  340 POKE USR "s"+3,BIN 11111111
  350 POKE USR "s"+4,BIN 01111111
  360 POKE USR "s"+5,BIN 00000001
  370 POKE USR "s"+6,BIN 00000011
  380 POKE USR "s"+7,BIN 00000111
  390 POKE USR "u"  ,BIN 00000000
  400 POKE USR "u"+1,BIN 10000000
  410 POKE USR "u"+2,BIN 11000000
  420 POKE USR "u"+3,BIN 11111000
  430 POKE USR "u"+4,BIN 00111111
  440 POKE USR "u"+5,BIN 11000000
  450 POKE USR "u"+6,BIN 10000000
  460 POKE USR "u"+7,BIN 00000000
  470 POKE USR "t"  ,BIN 00001000
  480 POKE USR "t"+1,BIN 00001000
  490 POKE USR "t"+2,BIN 00001110
  500 POKE USR "t"+3,BIN 00010000
  510 POKE USR "t"+4,BIN 01110000
  520 POKE USR "t"+5,BIN 01110000
  530 POKE USR "t"+6,BIN 01110000
  540 POKE USR "t"+7,BIN 10000000
  550 REM graficos ambulancia
  560 FOR n=1 TO 7: READ x: POKE USR "o"+n,x: NEXT n
  570 DATA 3,3,2,62,63,63,24
  580 NEXT n
  590 FOR n=1 TO 7: READ x: POKE USR "l"+n,x: NEXT n
  600 DATA 0,252,108,70,238,252,24
  610 NEXT n
  620 LET sd=20
  630 FOR m=1 TO 31
  640 PRINT INK 4;AT sd,m;"█"
  650 NEXT m
  700 REM edificios
  710 FOR c=2 TO 30
  720 FOR v=(7*RND)+12 TO 19
  730 PRINT INK 0;AT v,c;"█"
  740 PRINT OVER 1; INK 7; PAPER 0;AT v,c;" "
  741 NEXT v
  742 NEXT c
  750 LET r=0: LET i=0
  755 REM movimiento avion
  760 FOR a=0 TO 20 STEP 2
  770 LET y2=1
  780 FOR b=0 TO 30
  800 PRINT PAPER 5;AT a,b;" ": BEEP 0.001,40: PAUSE 1
  850 PAUSE 1: PRINT PAPER 5;AT a,b;"  "
  900 IF ATTR (a+i,b+r+3)=7 THEN LET r=r-2: LET a=a: IF a=a THEN GO TO 1000
  901 IF a=20 AND b=30 THEN PRINT FLASH 1;AT 10,10;"M U Y  B I E N": LET pt=pt*5: PRINT AT 20,0;" puntos totales";pt
  902 IF y2>=4 THEN GO TO 920
  905 IF INKEY$="0" THEN LET y2=y2+1: GO SUB 200
  920 LET a=a: LET b=b
  930 NEXT b
  940 NEXT a
 1000 FOR d=a TO 20: REM caida avion
 1015 BEEP 0.05,0.05
 1020 PRINT AT a,b;"  ": PAUSE 2: PRINT AT a,b;"  "
 1030 NEXT d
 1040 PRINT AT 20,b;"  "
 1050 RETURN
 1990 REM ambulancia
 2000 LET f=20
 2010 FOR g=30 TO b+2 STEP -1
 2020 PAPER 5: PRINT INK 3; OVER 1;AT f,g;" ": PAUSE 5: PRINT INK 4;AT f,g;"██"
 2025 PRINT INK 3;AT f,g;" ": BEEP 0.5,0.5: BEEP 0.3,0.5: PAUSE 1: PRINT PAPER 5;AT f,g;"  "
 2030 NEXT g
 2035 PRINT INK 3;AT f,g+1;" "
 2040 INPUT "otra vez ";s$
 2050 IF s$="s" THEN RUN
 2060 FOR n=1 TO 255: OUT 254,n: NEXT n: NEW
 3000 REM presentacion
 3020 PAPER 5: PRINT AT 10,1;"PULSE UNA TECLA PARA COMENZAR": PAUSE 0
 3030 CLS
 3100 PAUSE 25: RETURN
```



```
   10 POKE 23658,0
   20 RESTORE 9000: GO SUB 9000: REM GRAFICOS
   25 CLS
   30 GO TO 2000 : REM JUEGO
  499 REM MOVIMIENTO HELICOPTERO
  500 LET C$=INKEY$: IF C$<>"q" AND C$<>"a" AND C$<>"l" AND C$<>"0" AND C$<>"A" AND C$<>"L" THEN RETURN
  505 IF C$="l" OR C$="L" THEN GO TO 600
  510 PRINT AT PFN,1;A$(6);AT PFN+1,1;B$(6)
  520 IF C$="q" THEN LET PFN=PFN-1
  530 IF C$="Q" THEN LET PFN=PFN-2
  540 IF C$="a" THEN LET PFN=PFN+1
  550 IF C$="A" THEN LET PFN=PFN+2
  560 IF PFN<1 THEN LET PFN=1
  570 IF PFN>20 THEN GO SUB 600
  580 PRINT AT PFN,1;A$(1);AT PFN+1,1;B$(1)
  590 RETURN
  600 FOR V=1 TO NM
  605 IF PFN+1=F(V) AND C(V)=1 THEN GO TO 650
  610 NEXT V
  620 PRINT OVER 1; INK 6;AT PFN+1,4;D$
  630 PRINT OVER 1; INK 6;AT PFN+1,4;D$
  640 RETURN
  649 REM DESTRUCCION TANQUE
  650 PRINT AT PFN+1,4; INK 6; OVER 1;D$(1 TO P(V)-5)
  651 BEEP .001,40
  652 PRINT AT PFN+1,4; INK 6; OVER 1;D$(1 TO P(V)-5)
  660 INK 7: PRINT AT F(V),P(V);"*": BEEP .005,8: PRINT AT F(V),P(V);" ": BEEP .005,6: PRINT AT F(V),P(V);" ": BEEP .005,4
  670 LET C(V)=0: LET NMM=NMM+1: LET PUNTOS=PUNTOS+10
  675 PRINT #0;AT 0,19;"PUNTOS:";PUNTOS
  680 IF NMM<NM THEN RETURN
  699 REM PASO DE PANTALLA
  700 FOR V=10 TO 30: BEEP .05,V
  710 BORDER 2: BORDER 6: BORDER 1
  720 NEXT V
  730 LET PANT=PANT+1: LET PUNTOS=PUNTOS+FUEL*10+FUEL
  735 PRINT #0;AT 0,19;"PUNTOS:";PUNTOS
  740 IF PANT<=12 THEN CLS . GO TO 2145
  750 LET L$="  HAS CONSEGUIDO ELIMINAR LOS     INVASORES DE LAS DOCE TORRES.  EL ALTO MANDO,AGRADECIDO,TE CON-CEDE UNAS LARGAS VACACIONES"
  760 PRINT #0;AT 0,0;"                    ": CLS
  770 FOR B=1 TO LEN L$: BEEP .05,30: PRINT L$(B);: NEXT B
  780 PRINT AT 20,1;"PULSA UNA TECLA PARA CONTINUAR"
  790 PAUSE 0
  800 BEEP .05,40: BEEP .06,44: BEEP .1,42: GO TO 7000
 1500 GO SUB 500: REM MOVIMIENTO
 1510 REM RUTINA PRINCIPAL
 1520 LET FUEL=FUEL-1: PRINT AT 0,18;"  ";AT 0,18;FUEL: IF FUEL<=0 THEN GO TO 6500
 1530 LET FTT=INT (RND*FT)
 1540 FOR V=0 TO FTT
 1545 IF FTT=V THEN GO TO 1600
 1550 LET X=INT (RND*NM+1): IF C(X)=0 THEN NEXT V
 1555 IF C(X)=0 THEN NEXT V
 1560 PRINT AT F(X),1; INK 3; OVER 1;E$(2 TO P(x)): BEEP .001,30: BEEP .001,28
 1570 PRINT AT F(X),1; INK 3; OVER 1;E$(2 TO P(X))
 1580 IF F(X)=PFN THEN GO SUB 6000
 1590 NEXT V
 1600 GO TO 1500
 2000 REM INICIAR JUEGO
 2010 LET Z$="***LAS TORRES*******POR ALEJANDRO LABARGA Y JORGE LIZAR*******TU MISION ES DESTRUIR LOS TANQUES ENEMIGOS QUE SE ESCONDEN EN LAS GRIETAS DE LAS TORRES*********TUS CONTROLES SON:"
 2015 LET Z$=Z$+"----Q- PARA SUBIR----A-PARA BAJAR----L-PARA DISPARAR.PULSA SIMULTANEAMENTE CAPS SHIFT PARA MOVER MAS DEPRISA************ESPERA UNOS SEGUNDOS Y PULSA UNA TECLA PARA JUGAR***** "
 2020 LET HA=1
 2025 FOR P=31 TO 0 STEP -1: PRINT AT 10,P;Z$(1 TO HA): LET HA=HA+1: BEEP .05,30: PAUSE 1: NEXT P
 2030 FOR V=2 TO LEN Z$
 2040 IF v<LEN Z$-32 THEN PRINT AT 10,0;Z$(V TO V+31)
 2050 IF v>=LEN Z$-32 THEN PRINT AT 10,0;Z$(V+1 TO )
 2060 BEEP .05,30: NEXT V
 2100 DIM A$(6,3): DIM B$(6,3): LET A$(1)="   ": LET B$(1)="   ": LET A$(2)="   ": LET B$(2)="   ": LET A$(3)="   ": LET B$(3)="   ": LET A$(4)="   ": LET B$(4)="   ": LET A$(5)="   ": LET B$(5)="   ": LET A$(6)="   ": LET B$(6)="   "
 2102 DIM P(16): DIM F(16): DIM C(16): DIM G$(22,11): LET G$(1)="           ": LET NAVES=3: LET AF=0
 2105 LET M$="                    ": LET N$="                    "
 2110 LET D$="____________________________"
 2115 LET E$="------------------------------------"
 2116 LET F$="__/\__/\___/\__/\___/\___/\__/\_"
 2130 RESTORE 9000: GO SUB 9000
 2135 LET FUEL=99: LET PUNTOS=0: LET PANT=1: LET PFN=11
 2140 IF INKEY$="" THEN GO TO 2140
 2145 LET FUEL=99: LET NMM=0: GO SUB 3000+PANT*100
 2150 BORDER 1: PAPER 0: CLS
 2156 FOR V=2 TO 22: LET G$(V)="           ": NEXT V
 2157 FOR V=1 TO NM
 2158 LET G$(F(V)+1)=N$(1 TO P(V)-20)+M$(P(V)-19 TO 11)
 2159 NEXT V
 2160 FOR V=1 TO 22: PRINT AT V-1,21; PAPER 0; INK 6;TAB 21;G$(V): NEXT V
 2165 PRINT AT 21,0; INK 7;F$(1 TO 21)
 2170 FOR V=1 TO NM: PRINT INK 4;AT F(V),P(V);" ": NEXT V
 2180 INK 7: PRINT AT PFN,1;A$(1);AT PFN+1,1;B$(1)
 2210 PRINT #0;AT 0,1;"NAVES:";NAVES;AT 0,10; FLASH 1;AF;AT 0,19; FLASH 0;"PUNTOS:";PUNTOS
 2220 FOR V=1 TO PANT: PRINT AT 0,V-1;" ": NEXT V
 2230 PRINT AT 0,13;"FUEL:";FUEL
 2240 GO TO 1500
 3098 REM DEFINICION PANTALLAS
 3099 REM 1 PANT
 3100 LET NM=9: LET FT=2: RESTORE 3120
 3105 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3120 DATA 2,24,4,29,6,21,8,26,11,23,13,28,14,26,18,26,19,22
 3160 RETURN
 3199 REM 2 PANT
 3200 LET NM=9: LET FT=4: RESTORE 3220
 3205 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3220 DATA 4,25,6,23,8,24,10,26,13,24,15,25,16,22,18,24,20,26
 3280 RETURN
 3299 REM 3 PANT
 3300 LET NM=10: LET FT=4: RESTORE 3320
 3305 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3320 DATA 2,28,4,23,6,25,8,22,10,27,12,24,14,25,16,29,18,26,20,23
 3370 RETURN
 3399 REM 4 PANT
 3400 LET NM=10: LET FT=5: RESTORE 3420
 3405 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3420 DATA 2,28,5,25,8,23,10,27,12,25,13,24,14,23,15,22,19,26,20,30
 3458 RETURN
 3499 REM 5 PANT
 3500 LET NM=11: LET FT=5: RESTORE 3520
 3505 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3520 DATA 2,26,3,21,5,30,7,25,9,24,11,23,13,23,15,23,17,27,18,25,19,23
 3575 RETURN
 3599 REM 6 PANT
 3600 LET NM=12: LET FT=5: RESTORE 3620
 3610 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3620 DATA 2,24,4,30,5,29,6,28,8,30,9,29,10,28,12,30,13,29,14,28,17,27,19,21
 3660 RETURN
 3699 REM 7 PANT
 3700 LET NM=13: LET FT=6: RESTORE 3720
 3710 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3720 DATA 2,23,3,22,5,23,6,22,6,23,9,22,11,23,12,22,14,23,15,22,16,23,17,22,18,23
 3755 RETURN
 3799 REM 8 PANT
 3800 LET NM=14: LET FT=6: RESTORE 3820
 3810 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3820 DATA 3,29,4,29,6,29,7,29,8,27,9,27
 3825 DATA 11,27,12,27,13,25,14,25,16,25,17,25,18,23,19,22
 3860 RETURN
 3899 REM 9 PANT
 3900 LET NM=15: LET FT=7: RESTORE 3920
 3910 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 3920 DATA 2,27,3,26,4,25,5,24,6,23
 3925 DATA 8,27,9,26,10,25,11,24,12,23
 3930 DATA 14,27,15,26,16,25,17,24,18,23
 3940 RETURN
 3999 REM 10 PANT
 4000 LET NM=16: LET FT=8: RESTORE 4020
 4010 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 4020 DATA 2,22,3,23,4,22,6,22,7,23,8,22
 4025 DATA 10,30,11,29,12,28,13,27,14,26
 4030 DATA 15,25,16,24,17,23,18,22,19,21
 4040 RETURN
 4099 REM 11 PANT
 4100 LET NM=16: LET FT=10: RESTORE 4120
 4110 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 4120 DATA 3,23,4,24,5,25,6,26,7,27,8,28,9,29
 4125 DATA 10,30,11,29,12,28,13,27,14,26,15,25,16,24,17,23
 4130 RETURN
 4199 REM 12 PANT
 4200 LET NM=16: LET FT=12: RESTORE 4220
 4210 FOR V=1 TO NM: LET C(V)=1: READ F(V): READ P(V): NEXT V
 4220 DATA 2,23,3,23,4,23,5,23,6,23,7,23,8,23,10,25
 4230 DATA 11,25,12,25,13,25,14,25,15,22,16,22,17,22,18,22
 4240 RETURN
 5999 REM
 6000 REM RUTINA EXPLOSION
 6010 IF PFN>20 THEN LET PFN=20
 6020 FOR V=2 TO 6
 6030 PRINT AT PFN,1;A$(V);AT PFN+1,1;B$(V)
 6040 BEEP .04,20-V
 6050 NEXT V: LET FUEL=99: LET NAVES=NAVES-1
 6060 IF NAVES>0 THEN LET PFN=11: GO TO 2145
 6070 CLS : INK 7: GO TO 7000
 6500 FOR V=PFN TO 20
 6510 PRINT AT V,1;A$(1);AT V+1,1;B$(1)
 6520 BEEP .11,20
 6525 PRINT AT V,1;A$(6);AT V+1,1;B$(6)
 6530 NEXT v
 6531 LET PFN=20: GO TO 6000
 7000 CLS : LET K$="HAS CONSEGUIDO LA MEJOR PUNTUA- CION.GRABA TU NOMBRE PULSANDO    LAS TECLAS CORRESPONDIENTES       (MAXIMO 12 LETRAS)PULSA 0 PARA    CONTINUAR"
 7010 IF PUNTOS<=AF THEN GO TO 7070
 7020 FOR V=1 TO LEN K$: PRINT K$(V);: BEEP .05,40: NEXT V
 7030 LET T$=INKEY$: PRINT AT 7,2;PUNTOS;AT 7,10;T$: LET AF=PUNTOS
 7040 FOR N=1 TO 11
 7045 IF INKEY$="" THEN GO TO 7045
 7047 IF INKEY$="0" THEN GO TO 7070
 7050 LET T$=T$+INKEY$: PRINT AT 7,10;T$
 7055 IF INKEY$<>"" THEN GO TO 7055
 7060 NEXT N
 7070 PRINT AT 7,2;AF;AT 7,10;T$;AT 6,12;"EL MEJOR:"
 7080 PRINT AT 20,0;"**PULSA UNA TECLA PARA JUGAR**"
 7085 FOR v=0 TO 100: NEXT v
 7090 IF INKEY$="" THEN FOR V=1 TO NM: LET C(V)=1: NEXT V: GO TO 7090
 7095 LET NAVES=3
 7100 GO TO 2135
 8999 REM CREACION GRAFICOS
 9000 FOR v=USR "a" TO USR "u"+7: READ c: POKE v,c: NEXT v
 9010 DATA 8,20,34,65,128,0,0,0
 9015 DATA 0,0,0,255,0,0,0,0
 9020 DATA 0,0,0,0,128,64,32,31
 9025 DATA 0,0,0,0,1,2,4,248
 9030 DATA 0,14,22,254,12,62,85,62
 9035 DATA 0,2,128,192,127,65,128,0
 9040 DATA 0,171,1,15,255,255,63,1
 9045 DATA 0,171,0,240,136,132,194,254
 9050 DATA 1,0,15,15,0,0,0,0
 9055 DATA 252,144,252,255,0,0,0,0
 9060 DATA 0,0,32,16,12,14,7,7
 9065 DATA 0,0,0,16,56,185,251,255
 9070 DATA 0,4,40,112,224,224,192,192
 9075 DATA 7,7,14,12,16,0,0,0
 9080 DATA 255,255,126,62,60,28,24,8
 9085 DATA 240,252,224,96,32,16,8,0
 9090 DATA 133,16,66,20,1,68,16,66
 9095 DATA 17,68,144,0,66,16,65,4
 9100 DATA 119,119,0,238,238,0,119,119
 9105 DATA 0,0,0,0,0,0,24,60
 9110 DATA 60,24,0,0,0,0,0,0
 9115 RETURN
```

NOTAS GRAFICAS

```
A B C D E F G H I J K L M N O P
Q R S T U
```

CARGA Y ALMACENAMIENTO DE PROGRAMAS CON VELOCIDAD VARIABLE (y III)

Paco MARTIN y José María DIAZ

Con este artículo, se completa la trilogía que hemos dedicado a un tema especialmente interesante en código máquina: la posibilidad de elegir la velocidad de transferencia cassette-ordenador, o lo que es lo mismo, cómo conseguir programar más deprisa o más despacio que el standar permitido. Con ello, esperamos haber satisfecho a nuestros lectores y haber dejado claro un tema tan complejo y atractivo como el expuesto.



A. PERERA

```
BAUDIO  CALL  NUMERO        OBTIENE EL NUMERO DE LA LINEA DE COMANDO.
        PUSH  HL
        CP    ":"
        JP    NZ,ERRORC
        LD    HL,799        EL RANGO DEBE ESTAR COMPRENDIDO ENTRE 8ØØ Y
                            5ØØØ.
        SBC   HL,DE
        JP    NC,ERRORB
        AND   A
        LD    HL,5ØØØ
        SBC   HL,DE
        JP    C,ERROR6
        LD    DE,(PRES)     TOMA EL NUMERO DIVIDIDO POR 1Ø.
        LD    (BAUD),DE     PARA ASIGNAR LA VELOCIDAD
        CALL  INVAL         E INICIALIZA VALORES.
        POP   HL
        INC   HL
        LD    A,(HL)
        CP    """"
        JP    NZ,ERRORA
        INC   HL
        RET
;
NAME    LD    B,1Ø          SEÑAL "1Ø CARACTERES PARA EL NOMBRE".
        LD    DE,NOMBRE     DIRECCION DEL BUFFER DEL "NOMBRE".
PNAME   LD    A,(HL)
        CP    """"
        JR    Z,CMP
        LD    (DE),A        EL "NOMBRE" DEL BLOQUE QUEDA ALMACENADO.
        INC   HL
        INC   DE
        DJNZ  PNAME
        LD    A,(HL)
        CP    """"
        RET   Z
        SCF                 SEÑAL "NOMBRE DEMASIADO LARGO".
        RET
CMP     LD    A,9           MIRA SI EXISTE UN "NOMBRE".
        CP    B
        LD    A,32          SE COMPLETAN CON ESPACIOS LOS 1Ø CARACTERES.
LLENA   LD    (DE),A
        INC   DE
        DJNZ  LLENA
        RET
;
CPNAME  LD    A,(HL)        COMPRUEBA QUE "NOMBRE" FINALIZA
        CP    """"          CON COMILLAS.
        JP    NZ,ERRORF
        LD    A,#FF         SEÑAL "NOMBRE NO ESPECIFICADO".
        LD    (NOMBRE),A
```

```
;
SINTAX  INC   HL            ANALIZA QUE LA SINTAXIS ES CORRECTA Y BIFURCA A LA
                            SUBRUTINA CORRESPONDIENTE.
        LD    A,(HL)
        CP    13
        JR    Z,PPROG
        CP    2Ø2 ; 'LINE'
        JR    Z,PROGL
        CP    17Ø ; 'SCREEN$'
        JR    Z,SPANT
        CP    175 ; 'CODE'
        JR    Z,CODE
        JP    ERRORC
;
PPROG   LD    HL,#8ØØØ      SEÑAL "NO AUTOEJECUCION"
        LD    (DIRT),HL
TIPO    XOR   A             ASIGNACION DE VALORES PARA LA CABECERA.
        LD    (CABEC),A
        LD    DE,(23635)
        LD    HL,(23627)
        SBC   HL,DE
        LD    (INFC),HL
        LD    HL,(23641)
        SCF
        SBC   HL,DE
        LD    (LONT),HL
        SCF
        RET
PROGL   CALL  ENUM          TOMA EL VALOR NUMERICO SIGUIENTE AL TOKEN LINE; SI ES
                            MAYOR DE 9999, ERRORB
        LD    HL,9999
        SBC   HL,DE
        JP    C,ERRORB
        LD    (DIRT),DE
        CALL  TIPO
        RET
;
SPANT   INC   HL            MIRA QUE EL SIGUIENT CARACTER SEA 'ENTER'
        LD    A,(HL)
        CP    13
        JP    NZ,ERRORC
        LD    HL,16384      ASIGNA LOS VALORES NECESARIOS Y RETORNA
        LD    (DIRT),HL
        LD    HL,6912
        LD    (LONT),HL
INSA    LD    HL,#8ØØØ
        LD    (INFC),HL
        LD    A,3
        LD    (CABEC),A
        RET
```

```
;
CODE    CALL  NUMERO        TOMA LOS DATOS DE DIRECCION Y LONGITUD A LA VEZ QUE
                            CHEQUEA SINTAXIS
        LD    (DIRT),DE
        CALL  COMDAT
DEFLON  LD    A,(HL)
        CP    ","
        JP    NZ,ERRORC
        CALL  ENUM
        LD    (LONT),DE
        CALL  COMDAT
        JR    INSA
;
COMDAT  LD    A,D           CHEQUEA QUE SI EL NUMERO ES CERO, EXISTIA AL MENOS
                            UN ASCII 'Ø'
        OR    E
        RET   NZ
        DEC   HL
        LD    A,(HL)
        CP    "Ø"
        JP    NZ,ERRORC
        INC   HL
        RET
;
ENUM    CALL  NUMERO        TOMA EL ULTIMO NUMERO ESPECIFICADO
        CP    13            Y ANALIZA QUE EL SIGUIENTE SEA
        JP    NZ,ERRORC     'ENTER'; SI NO, ERRORC
        RET
;
NSINT   XOR   A             CONTINUA ANALIZANDO LA SINTAXIS DE VERIFY Y LOAD, BI-
                            FURCANDO A LA SUBRUTINA ADECUADA
        LD    (CABEC),A
        INC   HL
        LD    A,(HL)
        CP    13
        SCF                 SEÑAL "PROGRAMA BASIC"
        RET   Z
        LD    A,3           SEÑAL "BYTES"
        LD    (CABEC),A
        LD    A,(HL)
        CP    17Ø ; 'SCREEN$'
        JR    Z,LPANT
        CP    175 ; 'CODE'
        JR    Z,LCODE
        JP    ERRORC
;
LPANT   CALL  SPANT         ASIGNA VALORES PARA "LOAD" - "VERIFY" DE PANTALLA
        XOR   A
        RET
```

```
;
LCODE   INC   HL            ANALIZA SI SE HA ESPECIFICADO LA DIRECCION
        LD    A,(HL)
        CP    13
        LD    A,1           SEÑAL "DIRECCION NO ESPECIFICADA"
        RET   Z
        DEC   HL
        CALL  NUMERO        TOMA EL VALOR DE LA DIRECCION
        LD    (DIRT),DE
        CALL  COMDAT
        LD    A,(HL)
        CP    13
        LD    A,2           SEÑAL "DIRECCION ESPECIFICADA"
        RET   Z
        JR    DEFLON
;
NUMERO  LD    DE,Ø          INICIALIZA EL VALOR A CERO
CRNUM   INC   HL            APUNTA AL SIGUIENTE CARACTER
        LD    A,(HL)        TOMA EL DATO
        CP    [illegible]   ¿ES MAYOR QUE "9"?
        RET   NC            RETORNA SI ES ASI
        CP    48            ¿ES MENOR QUE "Ø"?
        RET   C             RETORNA SI ES ASI
        SUB   48            AJUSTA EL VALOR ASCII A DECIMAL
        PUSH  HL            PRESERVA EL PUNTERO
        LD    (PRES),DE     ALMACENA EL NUMERO CALCULADO
        EX    DE,HL         INTERCAMBIA VALORES
        CALL  MULT          OPERACION HL=HL*1Ø
        LD    D,Ø
        LD    E,A
        ADD   HL,DE         ASIGNA EL NUEVO VALOR, QUE NO DEBE SER MAYOR QUE
                            65535
        JP    C,ERRORB
        EX    DE,HL
        POP   HL
        JR    CRNUM
;
PRES    DEFW  Ø
;
MULT    ADD   HL,HL         HL=NUMERO*2
        JP    C,ERRORB
        LD    D,H           COPIA HL EN DE
        LD    E,L
        ADD   HL,HL         HL=NUMERO*4
        JP    C,ERRORB
        ADD   HL,HL         HL=NUMERO*8
        JP    C,ERRORB
        ADD   HL,DE         HL=NUMERO+NUMERO*2
        JP    C,ERRORB
        RET
;
INVAL   LD    DE,DATSAV     TOMA LA DIRECCION DEL DATA PARA AJUSTAR VALORES EN
                            LA OPERACION SAVE EFECTUANDO EL CALCULO A CONTINUA-
                            CION
        CALL  CALCUL
        LD    HL,BUFFER
        LD    A,(HL)        EL REGISTRO HL SE UTILIZA COMO PUNTERO DE LOS DATOS
                            OBTENIDOS.
                            INSERTA LOS VALORES ADECUADOS EN LAS POSICIONES CO-
                            RRESPONDIENTES PARA LA OPERACION "SAVE"
        LD    (DS1+2),A
        INC   HL
        LD    A,(HL)
        LD    (DS2+1),A
        INC   HL
        LD    (DS3+1),A
        INC   HL
        LD    A,(HL)
        LD    (DS4+1),A
        LD    DE,DATLOA     EFECTUA LA MISMA OPERACION PARA
        CALL  CALCUL        "LOAD", "VERIFY", "MERGE"
        LD    HL,BUFFER
        LD    A,(HL)
        NEG
        LD    (DL1+1),A
        INC   HL
        LD    A,(HL)
        NEG
        LD    (DL2+1),A
        INC   HL
        LD    A,(HL)
        NEG
        LD    (DL3+1),A
```

```
         INC    HL
         LD     A,(HL)
         NEG
         LD     (DL4+1),A
         RET
                                  ESTA SUBRUTINA EFECTUA EL CALCULO (DDAT)*150/(BAUD)
CALCUL   LD     B,4               SE VAN A CALCULAR 4 VALORES
         LD     HL,BUFFER         SE INICIALIZA EL PUNTERO DE
         LD     (DDAT),HL         DIRECCION DONDE SE VA A COLOCAR EL RESULTADO DEL
NCALC    PUSH   BC                CALCULO
         LD     A,(DE)            TOMA EL PRIMER VALOR A CALCULAR
         PUSH   DE
         LD     HL 15∅
         LD     D,∅
         LD     E,A
         CALL   #3∅ A9            OPERACION HL=HL*DE
         LD     B,H
         LD     C,L
         PUSH   BC
         LD     BC,(BAUD)         TOMA EL VALOR DE LOS BAUDIOS Y LO ALMACENA EN EL
                                  STACK
         CALL   #2D2B
         POP    BC
         PUSH   HL
         CALL   #2D2B
         POP    DE
         CALL   #31AF             SE DIVIDEN AMBOS VALORES
         CALL   #2DA2             RECUPERA EN A EL RESULTADO
         LD     HL,(DDAT)         EL RESULTADO QUEDA ALMACENADO
         LD     (HL),A
         INC    HL                EL PUNTERO SE INCREMENTA
         LD     (DDAT),HL
         POP    DE
         INC    DE
         POP    BC
         DJNZ   NCALC
         RET
;
DDAT     DEFW   ∅
BUFFER   DEFB   ∅,∅,∅,∅
;
BAUD     DEFW   15∅
;
DATSAV   DEFB   59,66,62,49
DATLOA   DEFB   8∅,78,53,8∅
;
CABEC    DEFB   ∅
NOMBRE   DEFB   127
         DEFM   "M. HOBBY"
LONT     DEFW   ∅
DIRT     DEFW   ∅
INFC     DEFW   ∅
;
BUFCAB   DEFB   ∅
BUFNAM   DEFS   1∅
LONBUF   DEFW   ∅
DIBUF    DEFW   ∅
INFBUF   DEFW   ∅                 SE EFECTUA LA MISMA OPERACION QUE EN LA ROM. SALVO
;                                 QUE COLOCAMOS NUESTROS PROPIOS VALORES CALCULA-
                                  DOS ANTERIORMENTE EN LOS LUGARES SEÑALADOS CON
                                  UNA "      "
;
;        LOAD
LOAD     INC    D
         EX     AF,AF'
         DEC    D
         DI
         LD     A,#∅ F
         OUT    (#FE),A
         LD     HL,#∅ 53F
         PUSH   HL
         IN     A,(#FE)
         RRA
         AND    #2∅
         OT     #∅ 2
         LD     C,A
         CP     A
LBREAK   RET    NZ
LSTART   CALL   #∅ 5E7
         JR     NC,LBREAK
         LD     HL,#∅ 415
LDWAIT   DJNZ   LDWAIT
         DEC    HL
         LD     A,H
         OR     L
         JR     NZ,LDWAIT
         CALL   #∅ 5E3
         JR     NC,LBREAK
LEADER   LD     B,#9C
         CALL   #∅ 5E3
         JR     NC,LBREAK
         LD     A,#C6
         CP     B
         JR     NC,LSTART
         INC    H
         JR     NZ,LEADER
LDSYNC   LD     B,#C9
         CALL   #∅ 5E7
         JR     NC,LBREAK
         LD     A,B
         CP     #D4
         JR     NC,LDSYNC
         CALL   #∅ 5E7
         RET    NC
         LD     A,C
         XOR    #∅ 3
         LD     C,A
         LD     H,#∅ ∅
DL1      LD     B,#B∅ :
         JR     MARKER
LDLOOP   EX     AF,AF'
         JR     NZ,LDFLAG
         JR     NC,VERIFY
         LD     (IX+∅ ),L
         JR     LDNEXT
LDFLAG   RL     C
         XOR    L
         RET    NZ
         LD     A,C
         RRA
         LD     C,A
         INC    DE
         JR     LDDEC
VERIFY   LD     A,(IX+∅ )
         XOR    L
         RET    NZ
LDNEXT   INC    IX
LDDEC    DEC    DE
         EX     AF,AF'
DL2      LD     B,#B2 ;
MARKER   LD     L,#∅ 1
LDBITS   CALL   #∅ 5E3
         RET    NC
DL3      LD     A,#CB :
         CP     B
         RL     L
DL4      LD     B,#B∅ :
         JP     NC,LDBITS
         LD     A,H
         XOR    L
         LD     H,A
         LD     A,D
         OR     E
         JR     NZ,LDLOOP
         LD     A,H
         CP     #∅ 1             DE LA MISMA MANERA QUE EN "LORD". INTRODUCIMOS
         RET                      NUESTROS PROPIOS VALORES
;
;        SAVE
;
SAVE     LD     HL,#∅ 53F
         PUSH   HL
         LD     HL,#1F8∅
         BIT    7,A
         JR     Z,SAFLAG
         LD     HL,#∅ C98
SAFLAG   EX     AF,AF'
         INC    DE
         DEC    IX
         DI
         LD     A,#∅ 2
         LD     B,A
SLEAD    DJNZ   SLEAD
         OUT    (#FE),A
         XOR    #∅ F
         LD     B,#A4
         DEC    L
         JR     NZ,SLEAD
         DEC    B
         DEC    H
         JP     P,SLEAD
         LD     B,#2F
SSINC1   DJNZ   SSINC1
         OUT    (#FE),A
         LD     A,#∅ D
         LD     B,#37
SSINC2   DJNZ   SSINC2
         OUT    (#FE),A
DS1      LD     BC,#3B∅ E ;
         EX     AF,AF'
         LD     L,A
         JP     SSTART
SALOOP   LD     A,D
         OR     E
         JR     Z,SPARYT
         LD     L,(IX+∅ )
LPARYT   LD     A,H
         XOR    L
SSTART   LD     H,A
         LD     A,#∅ 1
         SCF
         JP     SABITS
SPARYT   LD     L,H
         JR     LPARYT
SABIT2   LD     A,C
         BIT    7,B
SABIT1   DJNZ   SABIT1
         JR     NC,SAOUT
DS2      LD     B,#42 ;
SASET    DJNZ   SASET
SAOUT    OUT    (#FE),A
DS3      LD     B,#3E ;
         JR     NZ,SABIT2
         DEC    B
         XOR    A
         INC    A
SABITS   RL     L
         JP     NZ,SABIT1
         DEC    DE
         INC    IX
DS4      LD     B,#31 ;
         LD     A,#7F
         IN     A,(#FE)
         RRA
         RET    NC
         LD     A,D
         INC    A
         JP     NZ,SALOOP
         LD     B,#3B
SDELAY   DJNZ   SDELAY
         RET
```

Estimados lectores, por un error en la fotocomposición del listado en lenguaje ensamblador, que publicábamos en la segunda parte de este extenso artículo sobre carga con velocidad variable, insertamos a continuación una serie de líneas que fueron omitidas:

En la instrucción n.º 51 "HALT", hay que añadir la etiqueta "PSAV".

En la instrucción 145 "JP", se omitió "NZ,ERRORR".

En la instrucción 163 se puso "RES ∅,(IY+82)" y lo correcto es "RES ∅,(IY+2)".

En la 164 hay que poner "LD (IY+82),3".

CARGADOR DE LA RUTINA DE CARGA VARIABLE

```
   1 RESTORE : LET check=0: PRIN
T "CHEQUEO: ";
   2 FOR i=1 TO 1026: READ a: PO
KE 59999,a: LET check=check+a: N
EXT i
   3 PRINT check;" ";("in" AND c
heck<>154510);"correcto"
  10 DATA 42,93,92,35,126,254,23
4,32,30,35
  20 DATA 126,254,58,32,24,35,12
6,254,248,40
  30 DATA 28,254,239,202,27,235,
254,214,202,254
  40 DATA 234,254,213,40,82,24,2
,207,5,207
  50 DATA 9,207,10,207,11,207,14
,207,26,205
  60 DATA 254,235,205,45,236,56,
244,205,88,236
  70 DATA 245,62,253,205,1,22,17
5,17,161,9
  80 DATA 205,10,12,253,203,2,23
8,205,212,21
  90 DATA 221,33,234,237,17,17,0
,175,205,153
 100 DATA 238,6,50,118,16,253,24
1,237,91,245
 110 DATA 237,62,255,221,42,83,9
2,56,4,221
 120 DATA 42,247,237,205,153,238
,201,205,254,235
 130 DATA 205,45,236,220,76,236,
35,126,254,13
 140 DATA 32,167,205,174,235,237
,75,6,238,197
 150 DATA 3,247,54,128,235,209,2
29,229,221,225
 160 DATA 62,255,205,116,235,195
,205,8,205,9
 170 DATA 235,253,203,37,134,48,
117,24,91,205
 180 DATA 254,235,205,45,236,220
,76,236,205,241
 190 DATA 236,245,205,174,235,24
1,201,205,9,235
 200 DATA 253,203,37,198,48,88,2
37,91,6,238
 210 DATA 42,83,92,25,235,42,89,
92,55,237
 220 DATA 82,56,10,40,8,68,77,23
5,205,232
 230 DATA 25,24,11,25,235,167,23
7,82,68,77
 240 DATA 235,205,85,22,42,83,92
,237,75,10
 250 DATA 238,9,34,75,92,42,8,23
8,124,230
 260 DATA 192,32,7,34,66,92,253,
54,10,0
 270 DATA 237,91,6,238,221,42,83
,92,62,255
 280 DATA 253,203,37,70,40,1,55,
205,12,238
 290 DATA 216,195,143,234,167,32
,19,221,33,0
 300 DATA 64,237,91,245,237,42,6
,238,167,237
 310 DATA 82,194,143,234,24,218,
254,2,48,10
 320 DATA 221,42,8,238,237,91,6,
238,24,204
 330 DATA 221,42,247,237,32,2,24
,242,237,91
 340 DATA 245,237,24,190,221,33,
251,237,17,17
 350 DATA 0,175,55,205,12,238,48
,242,253,203
 360 DATA 2,134,253,54,82,3,33,2
34,237,14
 370 DATA 128,58,251,237,190,32,
2,14,246,254
 380 DATA 4,48,217,17,192,9,197,
205,10,12
 390 DATA 193,17,252,237,33,235,
237,6,10,126
 400 DATA 60,32,3,121,128,79,26,
190,35,19
 410 DATA 32,1,12,215,16,246,203
,121,32,180
 420 DATA 62,13,215,201,205,43,2
37,229,254,59
 430 DATA 194,139,234,33,31,3,23
7,82,210,137
 440 DATA 234,167,33,136,19,237,
82,218,133,234
 450 DATA 237,91,76,237,237,83,2
24,237,205,97
 460 DATA 237,225,35,126,254,34,
194,135,234,35
 470 DATA 201,6,10,17,235,237,12
6,254,34,40
 480 DATA 11,18,35,19,16,246,126
,254,34,200
 490 DATA 55,201,62,9,184,62,32,
18,19,16
 500 DATA 252,201,126,254,34,194
,141,234,62,255
 510 DATA 50,235,237,201,35,126,
254,13,40,15
 520 DATA 254,202,40,44,254,170,
40,59,254,175
 530 DATA 40,86,195,139,234,33,0
,128,34,247
 540 DATA 237,175,50,234,237,237
,91,83,92,42
 550 DATA 75,92,237,82,34,249,23
7,42,89,92
 560 DATA 55,237,82,34,245,237,5
5,201,205,232
 570 DATA 236,33,15,39,237,82,21
8,137,234,237
 580 DATA 83,247,237,205,115,236
,201,35,126,254
 590 DATA 13,194,139,234,33,0,64
,34,247,237
 600 DATA 33,0,27,34,245,237,33,
0,128,34
 610 DATA 249,237,62,3,50,234,23
7,201,205,43
 620 DATA 237,237,83,247,237,205
,220,236,126,254
 630 DATA 44,194,139,234,205,232
,236,237,83,245
 640 DATA 237,205,220,236,24,216
,122,179,192,43
 650 DATA 126,254,48,194,139,234
,35,201,205,43
 660 DATA 237,254,13,194,139,234
,201,175,50,234
 670 DATA 237,35,126,254,13,55,2
00,62,3,50
 680 DATA 234,237,126,254,170,40
,7,254,175,40
 690 DATA 8,195,139,234,205,161,
236,175,201,35
 700 DATA 126,254,13,62,1,200,43
,205,43,237
 710 DATA 237,83,247,237,205,220
,236,126,254,13
 720 DATA 62,2,200,24,159,17,0,0
,35,126
 730 DATA 254,58,208,254,48,216,
214,48,229,237
 740 DATA 83,76,237,235,205,78,2
37,22,0,95
 750 DATA 25,218,137,234,235,225
,24,226,0,0
 760 DATA 41,218,137,234,84,93,4
1,218,137,234
 770 DATA 41,218,137,234,25,218,
137,234,201,17
 780 DATA 226,237,205,162,237,33
,220,237,126,50
 790 DATA 207,238,35,126,50,240,
238,35,126,50
 800 DATA 246,238,35,126,50,5,23
9,17,230,237
 810 DATA 205,162,237,33,220,237
,126,237,68,50
 820 DATA 92,238,35,126,237,68,5
0,125,238,35
 830 DATA 126,237,68,50,133,238,
35,126,237,68
 840 DATA 50,138,238,201,6,4,33,
220,237,34
 850 DATA 218,237,197,26,213,33,
150,0,22,0
 860 DATA 95,205,169,48,68,77,19
7,237,75,224
 870 DATA 237,205,43,45,193,229,
205,43,45,209
 880 DATA 205,175,49,205,162,45,
42,218,237,119
 890 DATA 35,34,218,237,209,19,1
93,16,209,201
 900 DATA 0,0,0,0,0,0,150,0,59,6
6
 910 DATA 62,49,80,78,53,80,0,12
7,32,77
 920 DATA 46,32,72,79,66,66,89,0
,0,0
 930 DATA 0,0,0,0,0,13,10,8,255,
129
 940 DATA 129,129,129,129,0,0,0,
0,0,0
 950 DATA 20,8,21,243,62,15,211,
254,33,63
 960 DATA 5,229,219,254,31,230,3
2,246,2,79
 970 DATA 191,192,205,231,5,48,2
50,33,21,4
 980 DATA 16,254,43,124,181,32,2
49,205,227,5
 990 DATA 48,235,6,156,205,227,5
,48,228,62
1000 DATA 198,184,48,224,36,32,2
41,6,201,205
1010 DATA 231,5,48,213,120,254,2
12,48,244,205
1020 DATA 231,5,208,121,238,3,79
,38,0,6
1030 DATA 176,24,31,8,32,7,48,15
,221,117
1040 DATA 0,24,15,203,17,173,192
,121,31,79
1050 DATA 19,24,7,221,126,0,173,
192,221,35
1060 DATA 27,8,6,178,46,1,205,22
7,5,208
1070 DATA 62,203,184,203,21,6,17
6,210,128,238
1080 DATA 124,173,103,122,179,32
,202,124,254,1
1090 DATA 201,33,63,5,229,33,128
,31,203,127
1100 DATA 40,3,33,152,12,8,19,22
1,43,243
1110 DATA 62,2,71,16,254,211,254
,238,15,6
1120 DATA 164,45,32,245,5,37,242
,175,238,6
1130 DATA 47,16,254,211,254,62,1
3,6,55,16
1140 DATA 254,211,254,1,14,59,8,
111,195,222
1150 DATA 238,122,179,40,12,221,
110,0,124,173
1160 DATA 103,62,1,55,195,252,23
8,108,24,244
1170 DATA 121,203,120,16,254,48,
4,6,66,16
1180 DATA 254,211,254,6,62,32,23
9,5,175,60
1190 DATA 203,21,194,235,238,27,
221,35,6,49
1200 DATA 62,127,219,254,31,208,
122,60,194,213
1210 DATA 238,6,59,16,254,201
```

CONSULTORIO

Software

¿Hay en España alguna casa importadora de Software para el ZX Spectrum que trate directamente con el público? Si es así, ruego pongan dirección y teléfono.

Xavier PARIS RUBIO-Barcelona

☐ Efectivamente existen algunas casas de software que tratan directamente con el público; por ejemplo, DINAMIC SOFTWARE. Encontrará su dirección y teléfono en cualquiera de nuestras revistas.

Conectar periféricos

Creo saber que a los ordenadores se les puede conectar juegos (en cassettes), y tengo entendido que también se necesita un Microdrive.

—¿El ordenador no es suficientemente capaz de realizar lo que está programado en la cinta? o ¿es un aparato que sirve como amplificador de velocidad?

En caso de que sea uno u otro ¿dónde se conecta el juego y el microdrive al ordenador? o ¿se conecta el juego al microdrive y éste al ordenador?

—¿Los interfaces son adaptadores entre el ordenador y otra cosa?, si es cierto, ¿cuáles son las cosas que necesitan interface?

—¿El ordenador es capaz de utilizar sus teclas para jugar con algún juego que se compre? o ¿necesita comprar mandos suplementarios? y ¿dónde se conectan?

—¿Me podrían explicar qué significa «Programas en cartucho» (microdrive) para Spectrum?

Juan Carlos BOTELLA-Málaga

☐ A los ordenadores se les puede conectar directamente o mediante «interfaces», una gran variedad de periféricos, como floppy disks, impresoras y desde luego, cassettes. En estos últimos, puede introducirse cintas que contengan todo tipo de programas, bien sean de juegos o de utilidades. Lo que el Spectrum hace con ellas es cargar dentro de su memoria el programa que contienen y ejecutarlo acto seguido, o sea, que sí es capaz de bregar con ello sin ningún problema.

Sin embargo, existen adaptados a otro tipo de periférico para el Spectrum, el microdrive, y lógicamente, vienen en cartuchos de cinta específicos para ese fin. Estos últimos no podrían cargarse y ejecutarse desde una cassette.

Monitores de todo tipo

Me quiero comprar un monitor para un ZX Spectrum 16 K pero ¿hay que comprarse forzosamente el particular para Spectrum o puede ser otro?

Si es otro ¿puede conectarse directamente del ordenador al monitor o el Spectrum sólo se puede conectar al suyo particular?

Luis CORRAL-Córdoba

☐ No existe realmente, monitor específico de Spectrum, ya que hay varios modelos en el mercado que están previstos para conectarse directamente a nuestro ordenador favorito. Si no utiliza alguno de estos monitores, siempre es posible conectar otro modelo, pero ello le obligará a abrir el ordenador y hacer algunas modificaciones en la salida de vídeo. La persona que le venda el monitor, podrá darle más detalles sobre el modelo concreto en el que se haya fijado Vd.

Spectrum Plus

Me haríais un favor enviándome información acerca del nuevo modelo SPECTRUM PLUS. Muy agradecido por vuestros servicios.

Alberto BELART ESQUERRA-Barcelona

☐ El nuevo modelo de SPECTRUM PLUS es, internamente, análogo al llamado Spectrum 48 K. La única diferencia consiste en el teclado semiprofesional, bastante cómodo, y en el precio, sensiblemente más alto.

Una cinta que no «Runea»

Soy suscriptor de su revista que, hasta la fecha, recibo regularmente. La cinta de obsequio que me enviaron, cuya presentación es impecable, no «RUNEA». Se trata del programa UTILITIES. Ya me dirán qué solución tiene el asunto.

También les agradecería me informaran sobre su oferta de programas grabados. ¿Por qué del 1 al 14?; ¿no sería mejor para comprobar la calidad de las grabaciones del 1 al 4, por ejemplo? (Suponga que los números se refieren a los correspondientes a los ejemplares semanales de la revista).

Deberían publicar en cada número una «fe de erratas» (inevitables en toda publicación) del número anterior.

Alberto SAGADE MOURIZ-Bejar

☐ Si la cinta de utilidades tiene algún problema, mándenosla por correo y si efectivamente no carga, le mandaremos otra.

Por un error de imprenta, apareció un 14 en lugar de un 4.

La cinta contiene los programas de los números 1 al 4 inclusive.

Ordenador de informática

Me gustaría saber si hay alguna diferencia entre un ordenador de informática o uno como el Spectrum 48 K. Y si hay diferencia ¿por qué? ¿Cuánto vale un ordenador de informática?

Octavio IZQUIERDO REY-Barcelona

☐ Si entiende por ordenador de informática el típico ordenador de gestión utilizado a nivel de Empresa, la diferencia con un ordenador personal como el Spectrum, estriba fundamentalmente, en la capacidad de memoria (de 20 Mb en adelante) y gran velocidad de acceso, y cuyos precios superan fácilmente los dos millones de pesetas.

Problemas con el Spectrum

Recientemente he adquirido un ZX Spectrum de 48 K y, hasta ahora me funcionaba perfectamente; pero ahora me ocurre lo siguiente: cuando estoy tecleando un programa me aparece en la pantalla: 1.º datos que estaban en otra línea, si sigo tecleando el programa me aparece casi toda la pantalla con interrogaciones; y si sigo tecleando, el ordenador emite un sonido cada vez que pulso una tecla.

Quisiera preguntarles a qué es debido esto y cómo puede solucionarse, y si es algo grave.

Fernando GARCIA-Basauri (Vizcaya)

☐ Su problema se puede deber a dos causas diferentes:

La primera, y desgraciadamente la más probable, es que su Spectrum esté estropeado, por lo que no le quedará más remedio que enviarlo a reparar. La segunda posibilidad, radica en la forma de conectarlo. Si al enchufar el ordenador conectamos primero la clavija que va al Spectrum y luego la toma de corriente del transformador de forma brusca, es muy corriente que se produzca una sobrecarga temporal que provoque los extraños fenómenos a los que se refiere. Pruebe a enchufar el ordenador con el máximo cuidado posible y si no se le presenta el problema significa que, casi con toda seguridad, su Spectrum funciona correctamente.

Ampliación de memoria

Dispongo de un Spectrum 16 K y observo que la mayoría de las cintas existentes en el mercado e incluso las que regalan son de 48 K. ¿Qué debo hacer?; ¿se puede ampliar mi Spectrum 16 K a 48 K por medio de algún dispositivo?; ¿cuánto me costará. Una vez ampliado a 48 K ¿sirven las cintas 16 K?

Antonio EPELDE-Zumaia

☐ Puede usted ampliar su Spectrum de 16 a 48 K por un precio aproximado de 8.000 pesetas. una vez hecho esto, los programas de 16 K puede seguirlos utilizando.

Almacenamiento instantáneo

Ante todo quisiera felicitarles por su revista, pero también escribo para cuestionarles las siguientes preguntas:

¿Se pueden utilizar el ZX-Interface 2 y los ZX-cartuchos como un sistema de almacenamiento instantáneo y, en caso afirmativo, dónde se pueden adquirir cartuchos «vírgenes»?

¿Existe en la actualidad a la venta algún tipo de cartuchos de aplicaciones?

¿Cuánto cuesta cada cartucho y cuántos títulos hay?

Daniel BORRAS - Vigo

☐ El Interface 2 permite, entre otras cosas, utilizar cartuchos, pero son de ROM (memoria de sólo lectura) o de EPROM (memoria de sólo lectura programable); por tanto, no resultaría nada práctico utilizar estos últimos como sistema de almacenamiento instantáneo a nivel de base de datos o similar, dada la dificultad de programación que estos soportes conllevan. No obstante, si usted tiene la paciencia necesaria para construirse otro sistema operativo u otro lenguaje distinto del Basic Sinclair, sí podría emplearlos.

El precio de cada cartucho ronda las 8.000 pesetas y, que nosotros sepamos, no contienen programas de aplicación.

Más utilidades

Sería muy interesante —al menos para mí— que los programas y la revista se enfocara también hacia el campo de la gestión (ficheros y datos), pues a mi modo de entender, un ordenador, por muy pequeño que sea, debería estar enfocado a este tipo de funciones más que a los juegos.

Sin otro particular, atentamente:

A.E.B.

☐ Le agradecemos su sugerencia y aprovechamos para comentarle que esto se lleva a cabo en la medida de lo posible. Procuramos presentar programas de utilidades junto con los programas de juegos, aunque en la proporción que creemos refleja las preferencias mayoritarias de los usuarios.

Sonidos raros

Me gustaría saber cómo producir con mi ordenador sonidos raros (no notas musicales), así como también ritmos (a ser posible de batería). Me parece que será posible, ya que en algunos juegos ya he oído sonidos bastante raros y que no se parecían en nada a una nota musical.

También me gustaría saber cómo dibujar las letras que salen al principio de todos los programas de la cara b de la cinta que se da de demostración por la compra del ordenador. Yo he listado los programas y he podido ver que no están hechos por el modo «G» (Graphics), sino que está hecho por una o dos sentencias.

Antonio MASSOT - Andratx (Mallorca)

☐ Creemos que para producir los sonidos que a usted le interesan debe olvidarse del Basic y realizarlos en código máquina, mediante experimentación; los detalles concretos de cómo realizar esto, dada su dificultad y extensión, quedan más allá de esta sección de consultorio; le recomendamos que se dirija a tiendas especializadas y consiga libros que contengan la ROM desensamblada del Spectrum. En ellos, encontrará referencias claras y precisas de las rutinas ROM escritas para ese fin.

Manual de manejo

Me gustaría saber si existe el MANUAL de manejo del ZX SPECTRUM 48 K, en CASTELLANO, y si es así, ruego me comuniquen dónde es posible adquirirlo.

Miguel Angel ZAPATA LOBO Collado Mediano (Madrid)

☐ El manual de manejo del ZX Spectrum viene incluido al comprar su ordenador, totalmente traducido al castellano; que nosotros sepamos, no se vende suelto.

Sin embargo, las tiendas especializadas en informática están repletas de libros para el Spectrum adaptados a todos los niveles de complejidad; si no le es posible obtener el manual, tal vez alguno de éstos satisfaga sus necesidades.

TRUE VIDEO e INVERSE VIDEO

Tengo un Spectrum de 48 K desde el mes pasado y, poco a poco, voy aprendiendo su manejo. Aunque ya he hecho mis primeros programillas, aún me quedan, naturalmente, lagunas que no sé resolver. Bueno, mi consulta es la siguiente: ¿Para que sirven las funciones TRUE VIDEO e INVERSE VIDEO, que están en las teclas 3 y 4, respectivamente? También me preocupa cómo conseguir definir gráficos yo mismo, pues el manual no me ayuda demasiado.

Joan SANCHO VALERO - Barcelona

☐ Las teclas TRUE VIDEO e INVERSE VIDEO sirven para escribir en modo directo palabras inversas, es decir, intercambiar el papel y la tinta que el ordenador emplee en ese momento.

True Video restituye la situación normal.

Su segunda pregunta, acerca de la forma de definir gráficos, ya ha sido contestada en números anteriores de la revista.

Inteligencia artificial

Mi pregunta es la siguiente: ¿Cómo puede conseguirse que una máquina aprenda de sus propios errores?

Estoy interesado en conocer los principios básicos y la posible bibliografía existente en España.

Manuel MUÑOZ - Betanzos (La Coruña)

☐ La pregunta que usted nos plantea es la que preocupa a todos los laboratorios de investigación informáticos que se ocupan de la inteligencia artificial (IA).

Queda más allá de esta sección, debido a la extensión y complejidad del tema, una explicación detallada de los métodos que se pueden utilizar en IA, aun los más elementales.

Lamentablemente, apenas existe en nuestro país bibliografía sobre el tema y mucho menos en nuestro idioma. Habrá que esperar un tiempo hasta que esto se divulge o dirigirse directamente a importadores para obtener información acerca de IA.

SI NO QUIERE TECLEAR SUS PROGRAMAS, MICROHOBBY LOS GRABA POR USTED:

(cortar por esta línea)

SOLICITUD DE CINTAS DE PROGRAMAS

Cada mes ponemos a su disposición una cinta con todos los programas publicados en los cuatro números de dicho mes.

Deseo recibir **en mi domicilio** la(s) cinta(s) que indico a continuación, **al precio de 550 pts. cada una,** más 75 pts. de gastos de envío **certificado** por cada cinta solicitada.

☐ Números 1 al 4
☐ Del 5 al 8
☐ Del 9 al 12
☐ Del 13 al 16

NOMBRE .... EDAD ....
APELLIDOS ....
DOMICILIO ....
CIUDAD .... PROVINCIA ....
C. POSTAL .... TELEFONO .... PROFESION ....

Marco con una (X) en el casillero correspondiente la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre HOBBY PRESS, S.A. ☐ Giro Postal N.º .... ☐ Contra reembolso

TARJETA DE CREDITO: ☐ VISA VISA N.º .... ☐ MASTER CHARGE N.º ....



Fecha de caducidad de la tarjeta ....

Firma:

SI NO QUIERE TECLEAR SUS PROGRAMAS, MICROHOBBY LOS GRABA POR USTED:

Franqueo
Postal

HOBBY PRESS, S. A.

Apartado de Correos

n.º **54.062** (Apartados Altos)

MADRID

DE OCASION

• INTERCAMBIO programas para el Spectrum (poseo más de 300). Asimismo deseo intercambiar información con otros usuarios que se inicien en el lenguaje ensamblador del Spectrum (Código Máquina). Tengo abundante bibliografía. Interesados dirigirse a: Angel G. Magaz. C/ Postas, 1-3º. ASTORGA (LEON). Tfn.: (987) 61 54 35.

• VENDO Spectrum 48 K., por 32.000 ptas., casi nuevo, regalo programas. Lote de 4 Juegos-Spectrum («Mundo Mutante», «Atrapado», «Espacio Libre» y «Corre Caminos») por 1.900 ptas. Teresa Girones García. C/ Tabarca, 11-4° C. Alicante-03012.

• INTERCAMBIO programas para ZX Spectrum e instrucciones, tengo cerca de 500 programas y algo menos de instrucciones. Interesados llamar o escribir mandando lista a: Agustín Parbole Jabardo. Pardo Bazán, 12. Madrid-18016. Tfn.: (91) 413 97 28.

• VENDO impresora de 80 columnas con entrada Centronics. Utiliza papel normal y sistema de arrastre por tracción o fricción; posee varios tipos de letras. Tfno.: (91) 459 52 38. Preguntar por Francisco.

• SE VENDE ZX-81 comprado en Julio con todas las conexiones, libro de instrucciones en Castellano, fuente de alimentación, 16 K RAM pack, inversor de vídeo interno, joystick de pulsadores, y toda mi programoteca consistente en más de 50 programas, incluye las 12 primeras revistas ZX.

• USUARIO de Spectrum de 48 K desea ponerse en contacto con usuarios preferentemente en la provincia de Córdoba para cambiar programas. Mis títulos son muy variados incluyendo números 1. Francisco José Toledano. Tfn.: (957) 63 04 79.

• VENDO ZX Spectrum 48 K con gran cantidad de programas importados y nacionales, libros, cassette Horizontes, cables y transformador. Todo por 49.000 pts. Preguntar por Alvaro Durán Martínez. Tfno.: 952/49 00 45. Einca Arraijanal. Carretera de Caín s/n. Alhaurín el Grande (Málaga).

• VENDO transformador AC para 3, 6, 7.5 y 9 y a 220 V. y 125 V. Llamar al 34 83 59 (Zaragoza). Preguntar por Fernando.

• CAMBIO programas Spectrum 16/48 K. Francisco Frontiñan. C/ Escultor Ramirez, 22. Zaragoza. Tfno.: 38 08 34.